TU MICRO

# COMMODORE

N.º 10 · SEGUNDA EPOCA · N.º 9 375 PTAS · (IVA INCLUIDO)

## TURB MICRO COMMODORE-128.

## TABLETA O LAPIZ: ALTERNATIVAS GRAFICAS.

## SUPERBASE PARA PROFESIONALES.

## INTERFACE RS-232.








# TARDAREMOS EN VER ALGO PARECIDO.

En informática ya es difícil sorprender. Pues bien, Commodore lo ha conseguido con Amiga.

Más que una nueva generación de ordenadores, el Amiga de Commodore representa un nuevo concepto. Un ordenador que ofrece una serie de posibilidades y abre unos caminos que hasta hoy eran impensables.

Entre las novedades de este ordenador profesional destacan: la posibilidad de trabajar con un procesador de 32 bits y de 3 coprocesadores específicos, actuando los 4 al tiempo.

Además, Amiga dispone de pantallas y ventanas configurables con una resolución de 640 x 400 puntos escogiendo entre una paleta de 4.096 colores.

Trabaja con 4 canales independientes polifónicos con voz masculina y femenina, y por si fuera poco, Amiga de Commodore es el único ordenador multitarea que puede efectuar varios trabajos simultáneamente.

El precio también es importante. Por 330.000 ptas. ex. IVA, el Amiga incluye:

- Unidad central con 768 K RAM.
- Monitor de alta resolución en color y sonido.
- Unidad de discos de 880 K.
- Teclado profesional y ratón.

Por mucho que avance la investigación en el campo de los ordenadores, tardaremos en ver algo parecido.




Imagina siempre lo mejor.



**Microelectrónica y Control, S.A.** *Unico representante de Commodore en España.*
*c/ Valencia, 49-53 Tel.(93) 325 57 02 08015 Barcelona - c/ Ardemans, 8 Tel. (91) 246 37 36 28028 Madrid*



UN AÑO DE ROCK
EDICIONES NUEVA LENTE
SUPER-CHOLLO del Año
EL LIBRO
UN AÑO DE ROCK
EL DISCO O CASSETTE
"GRANDES EXITOS" DE HOMBRES G
(Con dos canciones inéditas)
Y UN SUPERPOSTER
Pídetelo
POR SOLO 950 Pts.
(IVA Incluido)
HOMBRES G
Grandes Exitos
INCLUYE 2 TEMAS INEDITOS
NUEVA LENTE

**Director:**
ANTONIO M. FERRER ABELLO
**Redactor Jefe:**
FERNANDO LOPEZ MARTINEZ
**Redacción:**
ANTONIO CARVAJAL
JOSE LUIS DE DIEGO
JUAN M. LOPEZ MARTINEZ
PABLO GARCIA MOLINA
IGNACIO BARCO LUENGO
ALFREDO SINDIN VALERO
FERNANDO ACERO MARTIN
**Colaboradores:**
JOSE LUIS M. VAZQUEZ DE PARGA
ANTONIO MANZANERA
**Secretaria de Redacción:**
PILAR MANZANERA AMARO
**Maquetación:**
CARLOS GONZALEZ AMEZUA
CARLOS TALLANTE
**Ilustraciones:**
ANTONIO PERERA
RAMON POLO
**Fotografía:**
EQUIPO GALATA
**Directora Publicidad:**
CARMINA FERRER
**Publicidad Madrid:**
BEGOÑA LLORENTE
Tel.: (91) 457 69 23
**Publicidad Barcelona:**
ISIDRO IGLESIAS
Avda. Corts Catalanes, 1010
Tel.: (93) 307 11 13
**Director de Producción:**
VICENTE ROBLES
**Directora de Administración:**
MARIA ANTONIA BUITRAGO
**Suscripciones:**
MARIA GONZALEZ AMEZUA
**Redacción, administración, publicidad y suscripciones:**
Plza. República Ecuador, 2.
28016 MADRID. Tel.: 457 94 24
Télex 49371 ELOC E
**Dirección para correspondencia:**
Aptdo. de Correos 61.294
28080 MADRID
TU MICRO COMMODORE es una publicación mensual de Ediciones INGELEK. 

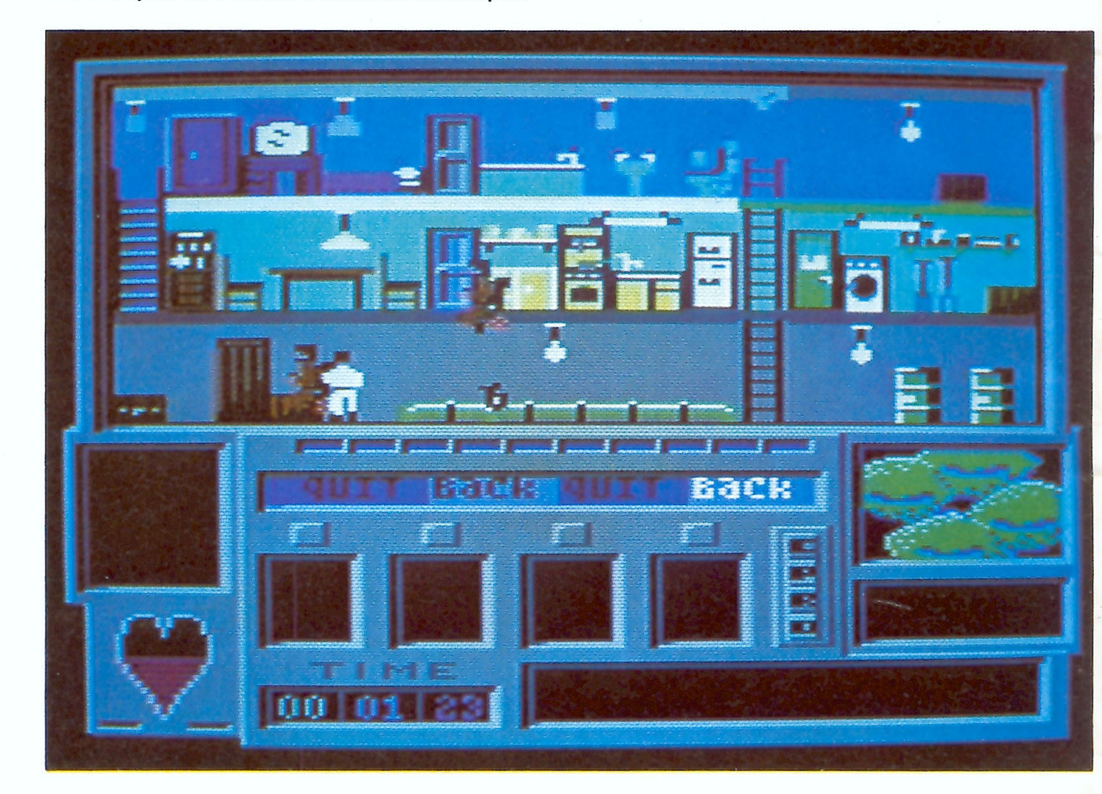
**Fotomecánica:**
RODACOLOR, S. A.
**Impresión:**
GRAFICAS REUNIDAS, S. A.
**Distribución:**
COEDIS, S. A.
Valencia, 245. Tel.: 215 70 97
08007 BARCELONA
Almacén: Nacional II. Km. 609,4
MOLINS DE REI (Barcelona)
Delegación en Madrid: Serrano, 165.
Tel.: 411 11 48
Almacén: Laforja, 19-21, esq. Hierro
Pol. Industrial Loeches
TORREJON DE ARDOZ (Madrid)

Precios para España. Ejemplar: 375 ptas. (IVA incluido). 355 ptas. (Canarias, Ceuta y Melilla).
(La suscripción anual incluye 11 números).
Distribución Cono Sur:
CADE, S.R.L.
Pasaje Sud América, 1532
Tel.: 21 24 64
Buenos Aires 1.290. Argentina.

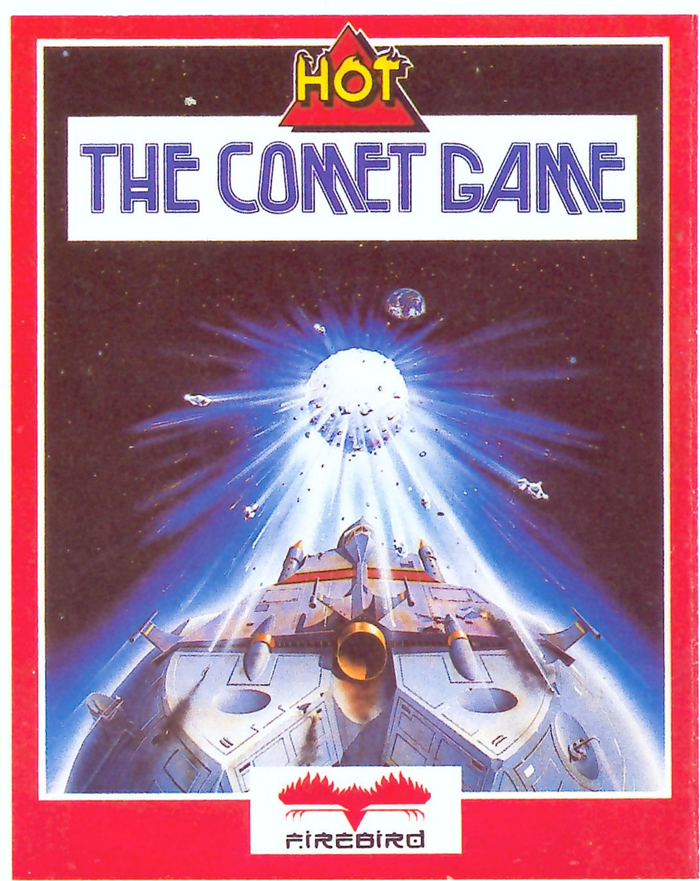
Depósito Legal: M. 40920-1985.
Impreso en España

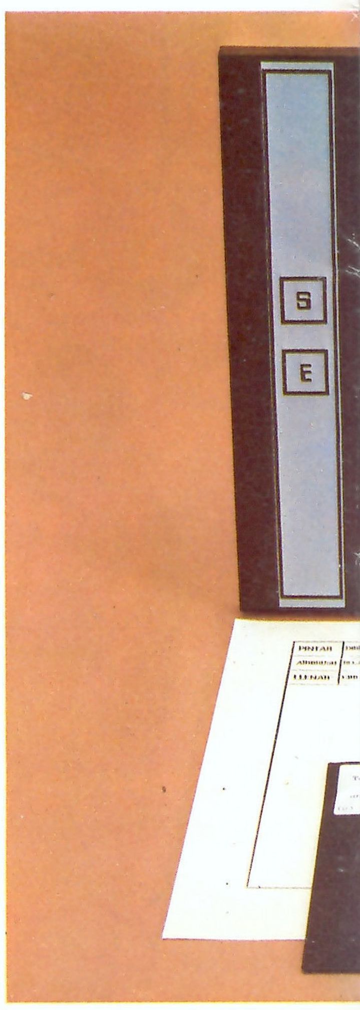
**enero 1987**

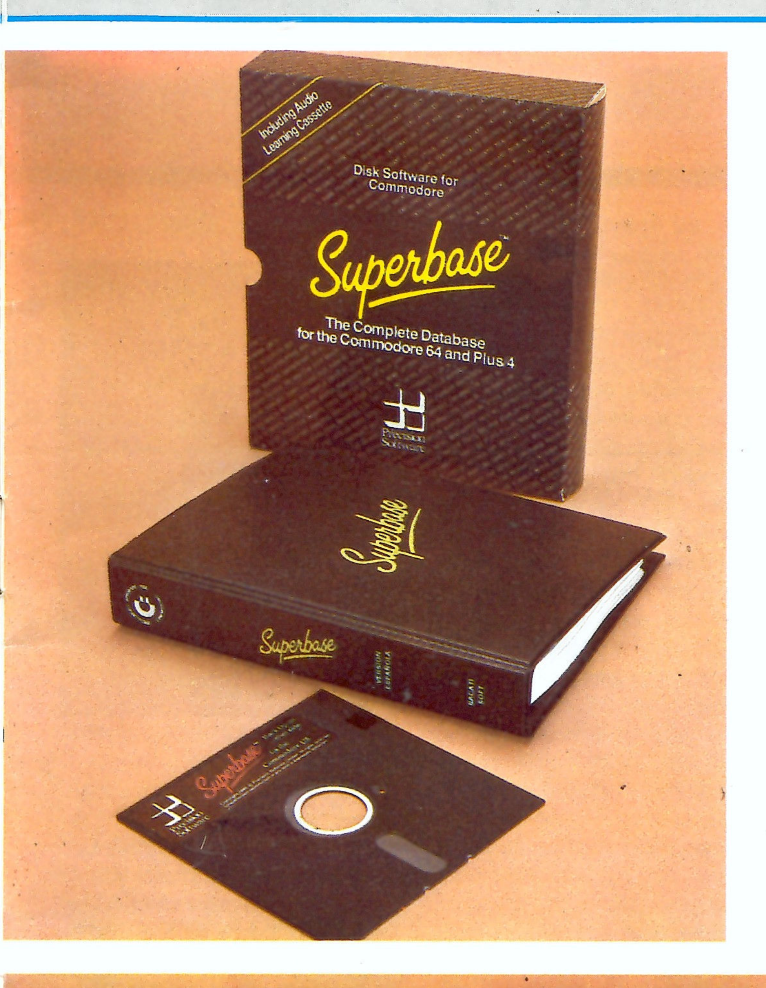
La eterna lucha entre el bien y el mal en el mundo del futuro. Recupera los Ocho Objetos de Carácter a través de los tiempos.



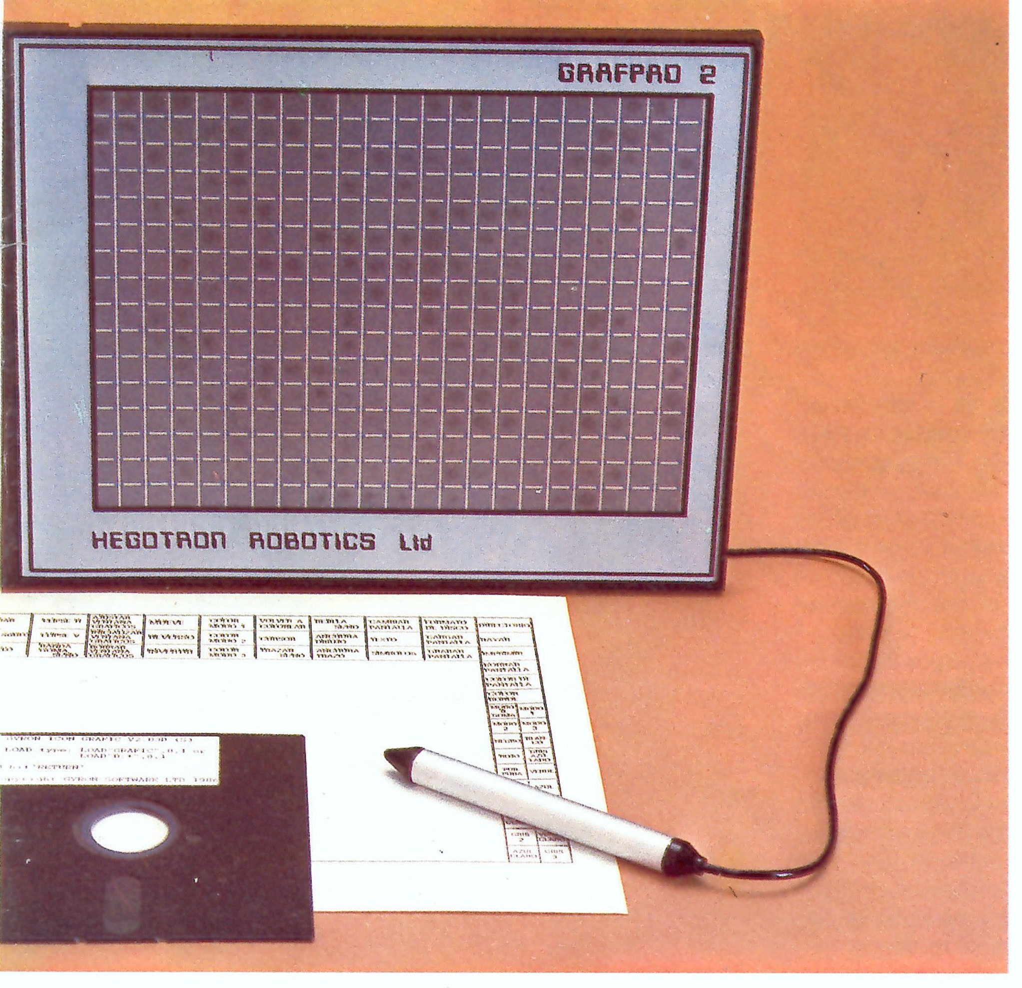

Multitud de misterios rodean al Cosmos y todo lo que en él se encuentra; Comet Game es la solución.






**SUPERBASE; potente Base de Datos para Commodore 64 y 128.**




**La tableta gráfica; un dispositivo diseñado para la construcción de gráficos en la pantalla.**

# SUMARIO

*Año Nuevo, vida nueva... y precios nuevos. Lamentamos tener que abrir este 1987 con una noticia nada agradable, aunque estamos seguros que todos comprenderéis que las circunstancias no dejan lugar a la elección. Cada día queremos hacer un TU MICRO COMMODORE más al gusto de todos, y a juzgar por vuestras opiniones lo vamos consiguiendo; no obstante, el papel que utilizamos, esos dibujos que tanto os gustan, la tinta que vertimos y como no ¡la necesidad de comer de los que la hacemos! son costes ineludibles. Brindemos pues por superar esta «barrera de los cinco duros al mes» (¡que es muy poco! ¿no?) y por un próspero y feliz año nuevo para todos.*

# NOTICIAS

### Interesante entrevista

Como ya adelantábamos en el número de diciembre, durante la celebración del SIMO, pudimos contactar con D. Josep Riera, perteneciente al departamento técnico de Microelectrónica y Control, quien amablemente puso luz a algunas de nuestras dudas.

### Primero Amiga

El primer tema abordado, fue la distribución de Commodore en España, y en concreto, las relaciones de MEC (Microelectrónica y Control) con Commodore USA.

En este sentido, se nos aclaró que el importador nacional tiene como contacto a Commodore de Inglaterra y no USA, como podría esperarse. Esto posibilita que en la actualidad, la presentación de los nuevos productos no tenga nada que envidiar al resto del mundo, dándose la paradoja, en el caso concreto del nuevo C-64, de su lanzamiento anterior en España que en Gran Bretaña.

Seguidamente, entramos de lleno en el fascinante mundo del Amiga, momento del cual extractamos ciertas conclusiones de gran interés.

El Workbench, sistema operativo de Amiga, es sin duda uno de los temas que despiertan un mayor interés en el usuario; máxime teniendo en cuenta las pequeñas deficiencias que se aprecian en su primera versión. En este sentido, se prevee la próxima aparición de la versión 1.2 de Workbench, con la idea de que no sólo corrija los errores de la anterior, sino que también prepare al equipo para futuras ampliaciones: control de Sidecard, coprocesador matemático, etc. No obstante, no se pudo determinar cuándo estaría disponible, ni cómo le será suministrada al usuario.

En lo referente a la política de precios, se tiene pensado que para el 87 el precio del Amiga disminuya, ya que Amiga no es sólo un ordenador sino una familia completa, con un equipo superior e inferior al que actualmente conocemos. Posiblemente, el Amiga más alto de la gama se comercializará al precio del modelo actual y éste a su vez bajará de precio; aunque este extremo no está aún confirmado, no deja de ser alentador.

Continuando por el sendero económico, pero esta vez a nivel empresarial, abordamos los rumores sobre los problemas financieros padecidos por la Commodore en Estados Unidos a partir de la aparición del Amiga.

En lo tocante a este tema, nuestro entrevistador matizó que MEC no se veía, en cualquier caso, afectada en absoluto por la supuesta crisis de la central norteamericana y que por otra parte, según la información de la que se dispone, dichos problemas se concretan a nivel de pagos de créditos que ciertos bancos concedieron a Commodore con la perspectiva de venta del Amiga, y no son al parecer tan graves como la prensa sensacionalista económica ha querido dar a entender.

Acerca del número de Amigas vendidos en España, no se nos ha podido proporcionar una cifra exacta, aunque sí es posible asegurar que la acogida que se la ha dispensado a este equipo es bastante aceptable, dadas sus peculiares características, que le encuandran en un grupo reducido de hobbistas, diseñadores gráficos, y programadores de grandes sistemas que aspiran a poner en práctica todos aquellos proyectos que no pueden realizar en su trabajo.

Sobre las posibilidades de introducción del Amiga en el campo de gestión, con objeto de presentar al mercado no sólo su capacidad de gráfica, cabe hacer hincapié en la ya mencionada ampliación de la gama Amiga, así como la aparición de Sidecar, que permite compatibilidad por software el equipo con un PC, previéndose la aparición de programas de facturación y contabilidad.

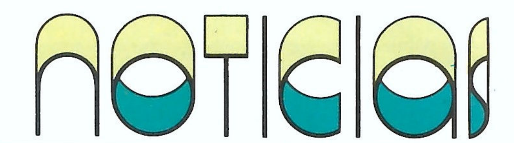


### La novedad: El sidecar

La opción de Sidecar se le presenta a todo usuario de Amiga interesado adicionalmente en la gestión.

Opera como un PC, emulando todos los gráficos (alfanuméricos y gráficos color y monocromo) en el mismo monitor. Incorpora un reloj interno, un controlador del DMA, 16K de ROM de BIOS y 128K de RAM para intercambiar información rápidamente con Amiga. Posee además una memoria de 256K RAM, ampliables a 640K, el almacenamiento se puede realizar en unidad de discos flexibles de 5.25" y 360K de capacidad, o en disco duro y unidades externas de 5.25" (360K) y 3.5 (720K) opcionales. La tarjeta trabaja con el sistema operativo MS-DOS 2.11 siendo necesaria la versión 1.2 de Workbench, y con un precio de





139.200 ptas. más IVA.

## PC-40 de Commodore

Otra de las últimas novedades que se añade a la línea actual del Commodore PC-10 y PC-20, es el Commodore AT o PC-40. El precio asignado de venta al público serán 685.000 ptas. más IVA y las primeras unidades se comercializaron a partir del mes de diciembre del pasado año.

Incorpora un procesador Intel 80286, trabajando a una frecuencia de reloj de 6 o 10 Mhz, con una disponibilidad de memoria RAM que se eleva a 1 Mb. En cuanto al almacenamiento masivo, dispone de una unidad de disco flexible de 1.2 Mb de capacidad y un disco duro de 20 Mb.

El equipo incluye un port paralelo Centronics y un serie RS-232C, así como 8 ranuras de expansión, totalmente compatibles con IBM PC/AT, de las cuales tres están ocupadas por el controlador de disco duro y floppy, adaptador de la placa gráfica y controlador serie/paralelo.

El teclado es completamente compatible con el de IBM PC/AT, separable y con 85 teclas y teclado numérico. En lo referente al monitor, se trata de un monocromo de 14", con resolución para el adaptador gráfico avanzado.

El sistema operativo empleado por el equipo es el MS-DOS 3.2, incorporando el lenguaje GW BASIC, un reloj-calendario controlado por el ordenador y llave de conexión.

## ¿Qué ocurre con la 1571?

También tuvimos ocasión de hablar con el Sr. Riera sobre la unidad de disco 1571, ya que hasta ahora no se encuentra muy difundida, a pesar de llevar más de un año en el mercado.

Según nuestro interlocutor, no hay motivo aparente para ello, y parece que todo se debe a que el usuario de C-128 se niega a entender que si quiere sacar el máximo rendimiento a su ordenador, la unidad de disco que necesita es la 1571 y no la 1541, por su velocidad, capacidad y posibilidad de trabajar en CP/M, entre otras cosas: «Como sabéis, el sistema operativo CP/M puede cargarse desde la d1541, pero pierde velocidad y capacidad, y lo que es más importante: el software de los ordenadores Osborne, Epson, Kaypro o el CP86 de IBM y compatibles no funciona más que en la 1571».

Desgraciadamente —continúa nuestro entrevistador— por una diferencia de 20.000 ptas., se pierden todas estas opciones, y luego, cuando el usuario comprende mejor su ordenador, se da cuenta y vienen las lamentaciones. Esto se debe principalmente a la falta de información, ya que a veces se compra a bulto, o por que un ordenador es más bonito que otro, aunque resulte increíble.







## El cambio de imagen

En el stand de MEC pudimos ver los nuevos modelos de unidad de disco 1541 y el C-64, que son exactamente iguales al modelo anterior, incluida la placa, cambiando sólo la apariencia externa para renovar la línea, pero compatibles por completo en software.

Tomando el camino de la difusión de Commodore, pudimos averiguar que el parque de ordenadores C-64 en todo el mundo asciende a más de 6 millones de equipos, siendo Estados Unidos el primero en ventas, seguido de Alemania e Inglaterra, ya que Alemania fabrica sus propios ordenadores y por tanto, el comprador lo considera como autóctono, lo mismo que le ha ocurrido a Inglaterra hasta hace poco tiempo, pues disponía también de una factoría. En España, la región más destacada en ventas es sin duda Cataluña.

## Enhorabuena a los ganadores

Este mes los agraciados ganadores de una suscripción por un año a TU MICRO COMMODORE, por la ayuda prestada en la confección de la sección LOS 7 MAGNIFICOS, han sido los siguientes lectores:

Sebastián Iglesias Carbonell de Sineu (Mallorca).

Alejandro Rodríguez Tormo (Valencia).

José Joaquín León Breviati (Sevilla).

Francisco J. Agis Miras (Granada).

Jordi Ramírez Magrinya (Barcelona).

La próxima ocasión puede llegarte el turno a ti: ¡anímate y mándanos tu voto!

# Camelot Warriors

De repente todo junto a ti se oscurece lenta, pausadamente, hasta que ni siquiera puedes apreciar qué objetos te rodean. Tu cuerpo dolorido se resiente por el peso de una ferrea armadura y de tu cintura pende una poderosa espada, larga y grave.

Poco a poco, una voz profunda penetra con sigilo en tu mente, como apoderándose de ti. Calmadamente, la voz te susurra frases, al parecer incoherentes: «Guerrero de Camelot... busca los cuatro objetos... el fuego que no quema... el espejo de la sabiduría... el elixir de la vida... la voz de otro mundo..., búscalos, morirás si no lo haces...»

Tu corazón palpita cada vez más deprisa, en parte por el miedo, en parte también porque sabes a qué objetos se refiere la voz, sabes cuál es tu misión, sabes dónde te encuentras. De alguna manera lo sabes y ya el ambiente en torno tuyo te resulta extraordinariamente familiar, sí, el lugar no lo conoces, aunque hay algo en él, que ya has visto cientos de veces, algo inexplicable...

Gran Bretaña, siglo VI aproximadamente. El Rey Arturo, desde Camelot, dirige los destinos del reino ayudado firmemente por sus fieles caballeros de la Tabla Redonda, quienes deben enfrentarse a un mundo de intrigas y magia, donde la vida de cada guerrero depende de su valor, de su destreza en combate, de su ingenio, de su capacidad para no aterrarse ante las más escalofriantes situaciones.

Dormías sumido en un sueño premonitorio, al menos hasta que el ruido de unos golpes en la puerta te obligaron a incorporarte. Son las doce de la noche y Su Majestad ha enviado a un escudero en tu busca; nada bueno cabe esperar de tan intempestiva llamada. En efecto, cuatro objetos del futuro —te dice el Rey Arturo— han penetrado en Camelot. La misión que como Rey tuyo te ordeno cumplir consiste en apoderarte de estos cuatro objetos, dado que ponen en peligro la seguridad de todo el reino.

Marcha raudo en dirección al bosque, pues he sabido que allí se encuentra el primero de todos ellos: EL FUEGO QUE NO QUEMA. Encuéntralo, no falles y encuéntralo, después busca a AZNATH, el druida amo del bosque, quién te dirá qué hacer a continuación. Parto veloz pues, Majestad. Al cabo de una hora ya habías llegado al milenario roble del bosque donde...

## FICHA TECNICA

| | |
|---|---|
| NOMBRE | CAMELOT WARRIORS |
| PRECIO | 2.300 PTAS. |
| SOPORTE | CINTA |
| TIPO | ACCION |
| MODELO | C-64, C-128 |
| OBSERVACIONES | |

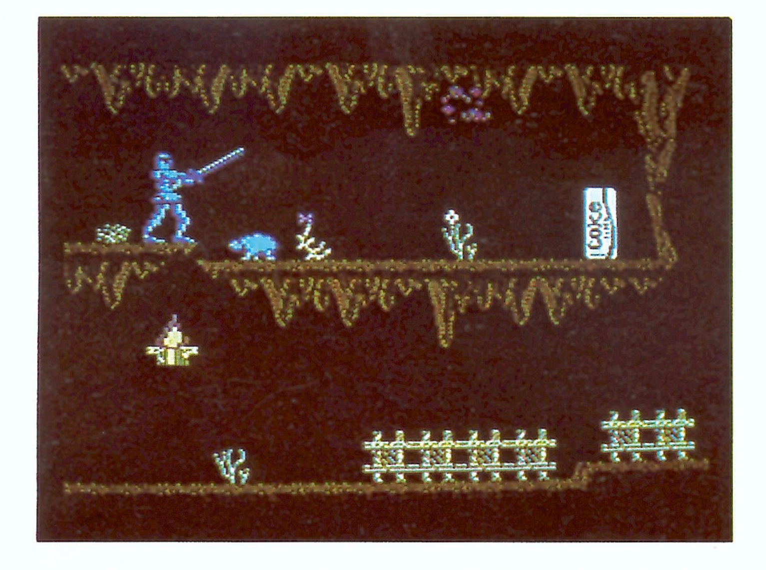
## *El bosque del druida Aznath*

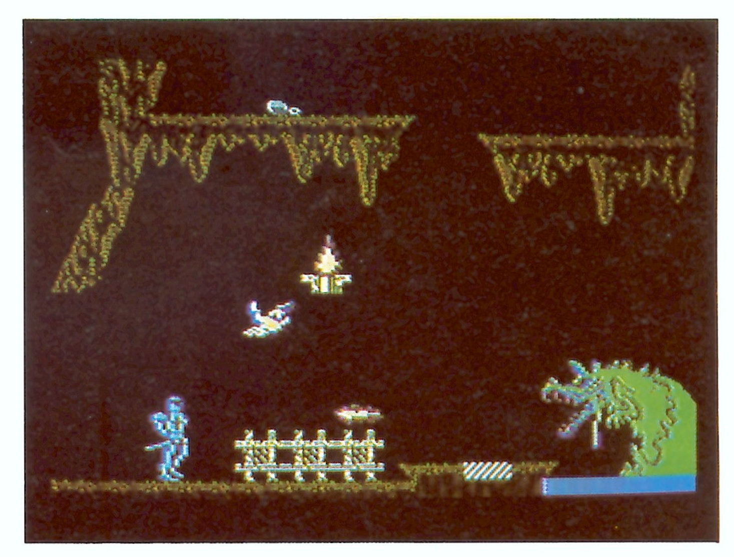
...donde tres extrañas criaturas se ciernen, pavorosamente sobre ti. Un golpe de espada, un salto huidizo y, de nuevo, todo está en calma. Frente a ti, un sendero se abre interminable y sombrío, mientras a lo lejos presientes extrañas criaturas arrastrándose. Temeroso, caminas con prisa a través del sendero hasta llegar a una abrupta grieta y sin pensar siquiera, sin mirar hacia atrás, saltas siempre hacia adelante cinco veces consecutivas. La muerte te ronda los talones.

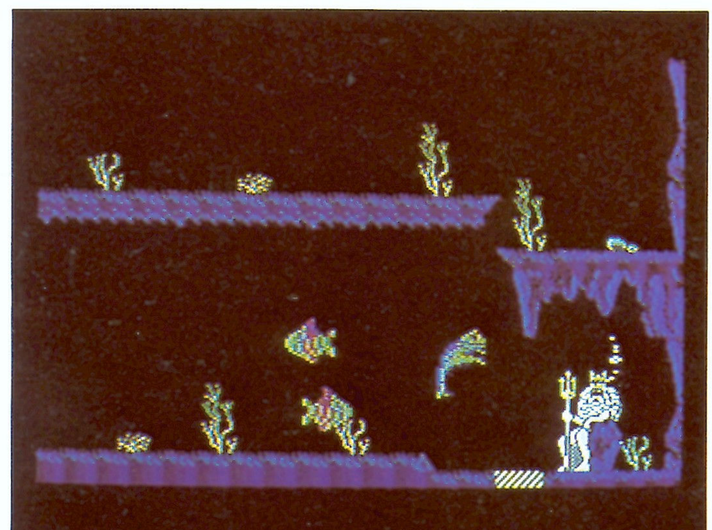
Gracias a tus agudos reflejos adviertes la presencia de un ser alado que clama por tu sangre, media vuelta espada en mano y seguirás vivo. Adelante, adelante, sólo la precaución te permitirá llegar hasta el FUEGO QUE NO QUEMA y, vencidos tus temores, asirlo con fuerza entre las manos. Paradójicamente ahora un mundo oscuro alberga al FUEGO QUE NO QUEMA, pero este extraño objeto ilumina la existencia de millones de hombres del siglo XX, aunque tú, desde luego, no puedes saberlo.

Como AZNATH te espera, deslízate por el abismo que custodian en su vuelo dos animales, recorre de nuevo el sombrío sendero y resbala con cuidado al término de éste. Mortales son las caidas, no obstante tu recia armadura las soporta perfectamente; al artesano que la forjó, debes tu vida. Ya en las zonas más profundas del bosque hallarás a tu izquierda al druida, distraido en sus experimentos de alquimia.

Te aproximas confiado hacia AZNATH, quien sigilosamente levanta los brazos y contra ti lanza un poderoso conjuro. Tus manos son verdes y viscosas, tu cuerpo entero es verde y viscoso: has sido convertido en un enorme batracio. Para acallar tus quejas, el druida te insta a marchar hacia el lago de Camelot. Sigue el camino de tu derecha y, después de saltar tan lejos como te sea posible, adéntrate en el lago —te ordena AZNATH tajantemente—.









## *El lago del Rey Kindo*

Hace ya mucho, mucho tiempo, tanto que ni las crónicas lo registran, Kindo, hermano del poderoso Neptuno, rey de todos los mares y océanos, tomó posesión del lago de Camelot, una extensa balsa de agua poblada por medusas, actinias y peces de gran tamaño. Aunque Kindo ha prohibido la entrada a su reino bajo pena de muerte, tu misión prima muy por encima de tu propia vida y como caballero has de asumir cualquier riesgo.

Nada más entrar en el lago observas como una medusa y una actinia de aspecto inocente, vigilan celosamente el camino. Para observar mejor, tratas de subirte a una pequeña piedra, mientras las dos criaturas acuáticas cubren su recorrido constante de un extremo a otro. ¡Ahora! no lo piensas, un salto, unos pasos y de nuevo, otro salto... casi te alcanzan. Enormes peces te cierran ahora el paso, mas afortunadamente aún no han detectado tu presencia; si consigues esquivarlos tal vez llegues, no sin algunas dificultades, hasta EL ESPEJO DE LA SABIDURIA.

El segundo objeto del futuro obra ya en tu poder. Retrocedes, pues las angostas cavidades acuáticas no te permiten avanzar hasta aquel precipicio, ¿recuerdas?, en la entrada del lago. No hay otro camino, ningún otro, salvo dejarte caer hacia las profundidades tan rápido como te sea posible. Y así, siguiendo el camino de tu izquierda, luego de descender por otro precipicio, llegas al fin hasta el fondo del lago.

Parejas de grandes peces, íntimos aliados de Kindo, son todo cuanto te separa del rey submarino. Algunos pasos y grandes saltos para que Kindo y tú os situéis frente a frente, cada uno sosteniendo la desafiante mirada del otro. Tus ancas ya casi no te sostienen en pie debido al cansancio, cuando le entregas EL ESPEJO DE LA SABIDURIA a Kindo. Has cerrado el ciclo del lago; por ello, recuperas tu antigua apariencia, apuesto guerrero de Camelot. Mas, de pronto, bajo tus pies cede el suelo y caes, caes, caes...

## *Las grutas del dragón Azornic*

La claridad de las cuevas donde has ido a parar, tenuemente iluminadas por antorchas, te obliga a cerrar los ojos. En pocos segundos ya puedes apreciar el nuevo mundo de grutas y cavernas que te rodea, el mundo de AZORNIC, el poderoso dragón, de quien dicen los ancianos de Camelot que jamás ha sido derrotado por hombre o demonio alguno. Cuentan también las leyendas de Camelot que la antesala del infierno son estas grutas, pobladas de espantosas alimañas en sus mil recovecos.

Se abre a tu izquierda un precipicio de dimensiones considerables, así que decides encaminarte hacia la derecha, precavido ante posibles ataques. Allí está, EL ELIXIR DE LA VIDA que debes entregar a AZORNIC, siempre y cuando las criaturas de las grutas lo permitan y, desde luego, sabes bien que intentarán por todos los medios a su alcance no permitirlo.

Al final de un despeñadero, una planta carnívora abre y cierra sus fauces, como es-

perando a su próxima víctima, saboreándola por anticipado. La distancia a cubrir te asusta tanto o más que la planta, por lo cual decides asomarte lo más posible y probar suerte. Todo ha salido bien y apenas si te separa medio milímetro del carnívoro vegetal; saltarás de nuevo, pero antes, comprobarás tu posición con el fin de no rozar la planta al tomar impulso.

Atraviesas ahora las galerias de la cueva con pasos firmes, dado que la luz es más clara y todo está tranquilo, tal vez demasiado tranquilo. Por momentos, el suelo se hace cada vez más impracticable, lleno de rocas que has de sortear. Tras el primer salto y tras dar unos pasos un animal se abalanza sobre ti desde la izquierda, arrastrándose, y otro desde la derecha, volando. ¿Qué hacer?, no puedo con ambos a un tiempo, prefiero morir espada en mano, morir como el más noble de los caballeros.

Nada surcaba ya el aire debido a tu bien ejecutado mandoble, aunque tu muerte parecía segura. De pronto, inexplicablemente, éste tu segundo enemigo, no pone fin a tu existencia, ni siquiera te derriba, se limita a pasar entre tus piernas sin dañarte lo más mínimo. Incapaz de comprender, prosigues tu viaje hasta la última piedra del estrecho pasadizo. Un riachuelo, además de un túnel vigilado por una extraña y diabólica criatura, y sus no menos extraños y diabólicos murciélagos, es todo cuanto te separa de AZORNIC.

AZORNIC, el más inteligente y poderoso de los dragones, huraño de carácter y malintencionado de espíritu, recibe con agrado el presente que le ofreces. Si hubieras tardado un poco más en ponerlo a sus pies, unos segundos más a lo sumo, habrías perecido bajo el fuego de AZORNIC. No ha ocurrido así, por tanto el señor de las grutas, en agradecimiento, te teletransporta mediante un hechizo hasta el castillo de Camelot.

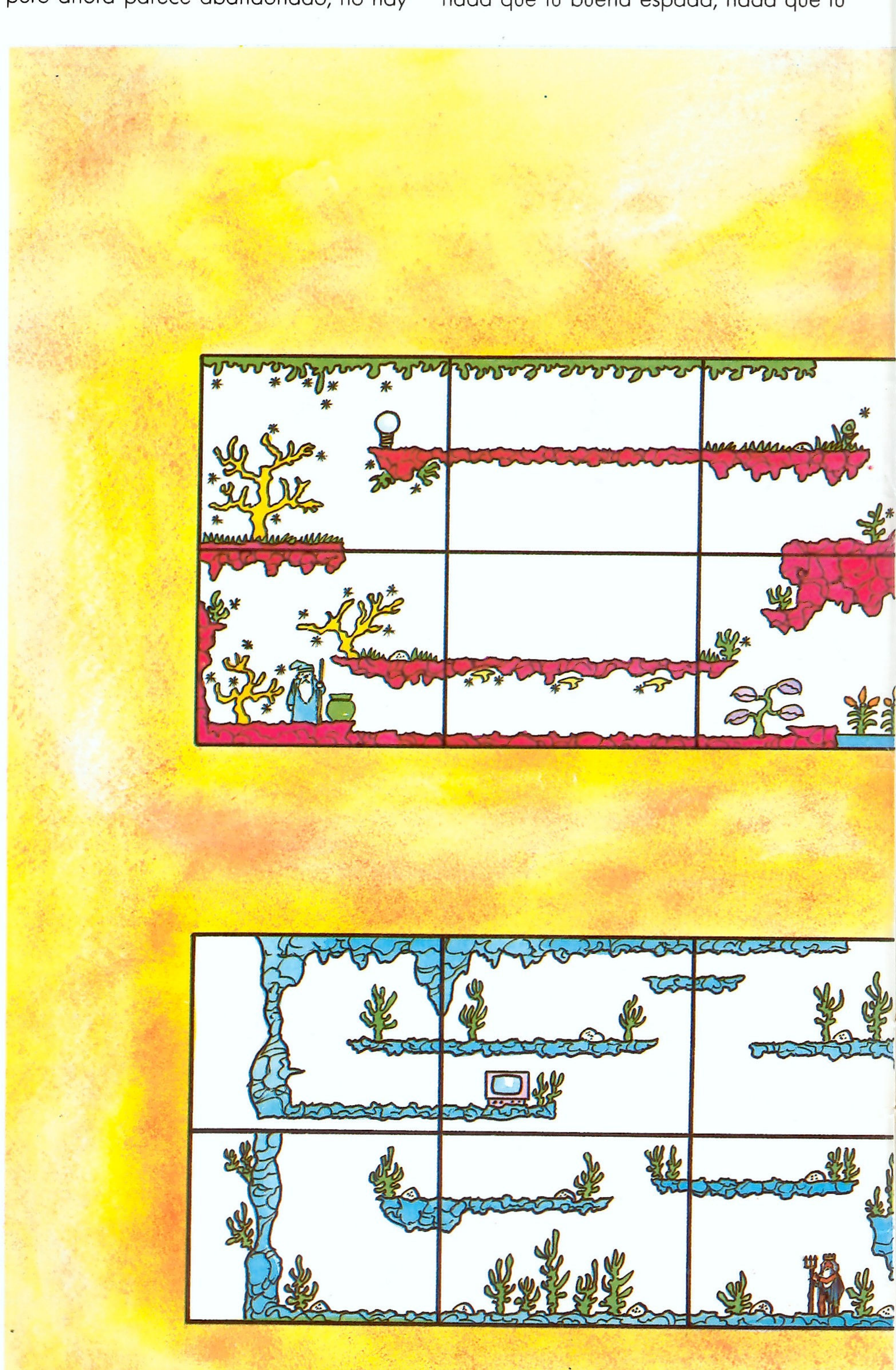
## El castillo del Rey Arturo

«Sede de la famosa tabla redonda, el castillo de Camelot, residencia permanente del Rey Arturo de Inglaterra, es la obra arquitectónica más impresionante del mundo conocido. Sus murallas, baluartes, fosos y demás fortificaciones, proporcionan una total seguridad al recinto interior del mismo». Siglo VI. Anónimo.

Recuerdas con toda claridad esta inscripción, sita en la fachada de la edificación, justo encima del pórtico principal. Pero también recuerdas que del antiguo esplendor del castillo queda muy poco ya, pues la desidia y la suciedad han atraido a muchos animales. La verdad sea dicha, y con el debido respeto a Su Majestad, este castillo no es habitable.

Rondan a través de sus tres plantas, dos de ellas separadas por medios techos, fantasmas, ratas, aves maléficas y desconocidas alimañas; aunque poco a poco los habitantes del castillo han aprendido a convivir con semejante compañia. Al salir en cumplimiento de tu misión, pese a lo intempestivo de la hora, una cierta actividad llenaba todas las habitaciones del castillo, pero ahora parece abandonado, no hay nadie.

La voz de tu sueño, LA VOZ DE OTRO MUNDO te llama desde lo alto de la torre. Subes deprisa y allí lo ves, el extraño objeto del que surge una voz apagada. Desciendes de la torre cayendo desde el peldaño más alto de ésta: como siempre, tu armadura protege tu cuerpo. Luego, varios pasillos y habitaciones, escaleras y diversos animales diseminados por todo el castillo. En fin, nada que tu buena espada, nada que tu



rapidez y nada que tu ingenio, caballero de CAMELOT, no pueda superar.

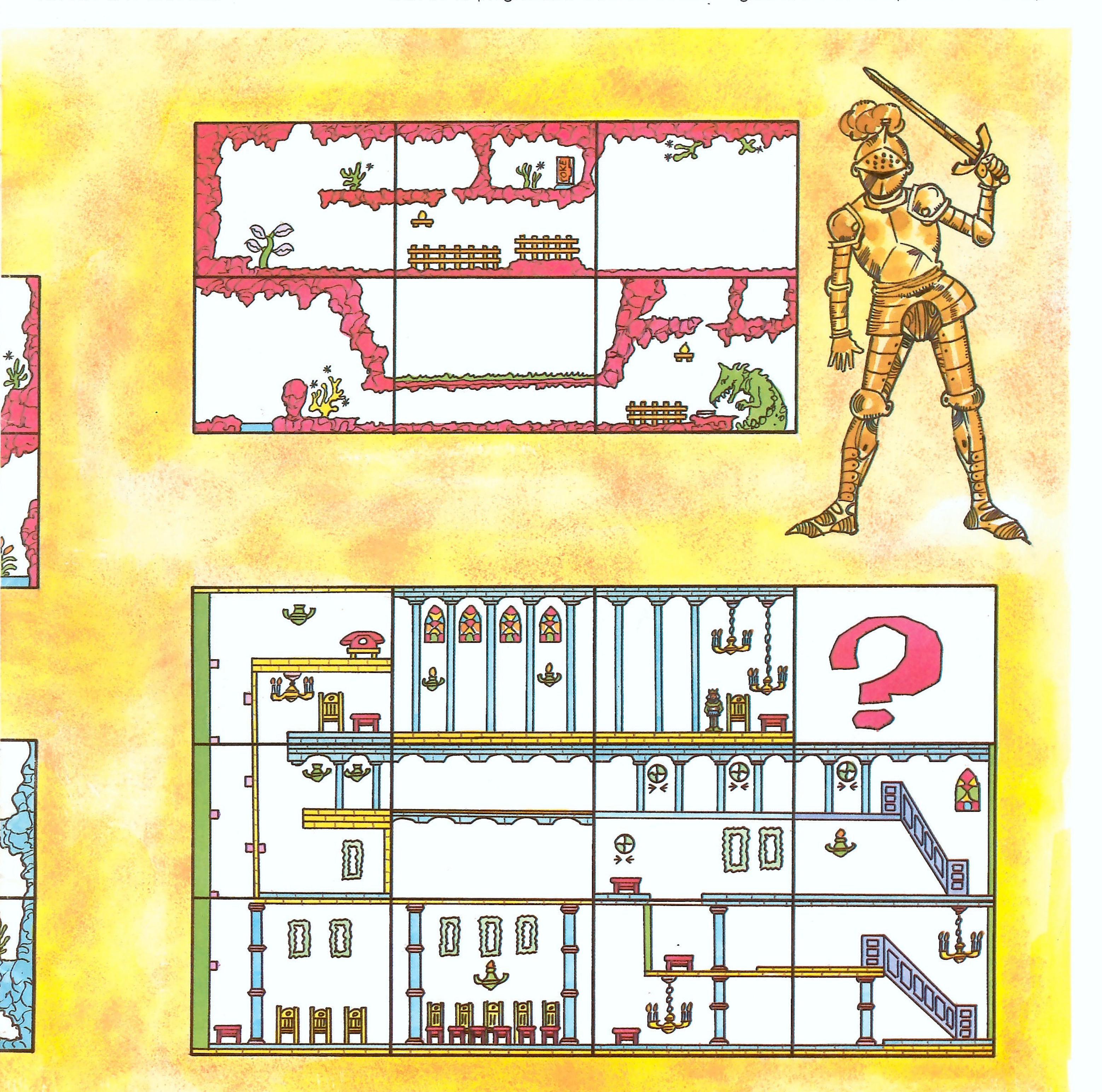
Por fin, has completado con éxito esta dura empresa al entregar a Su Majestad, el Rey Arturo, el último de los cuatro objetos del siglo XX. Sin fuerzas, comienzas a desvanecerte y mientras la oscuridad lo invade todo en torno tuyo, escuchas las palabras de tu monarca: la razón de esta descabellada empresa está en tu mente, el por que tan solo tú lo conoces...

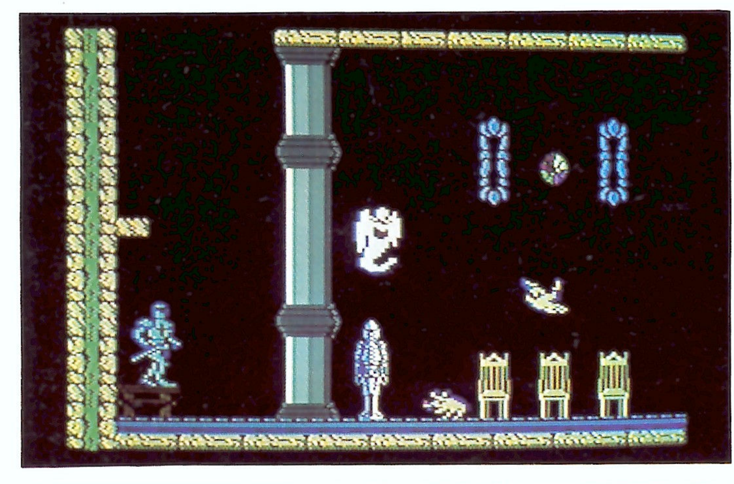
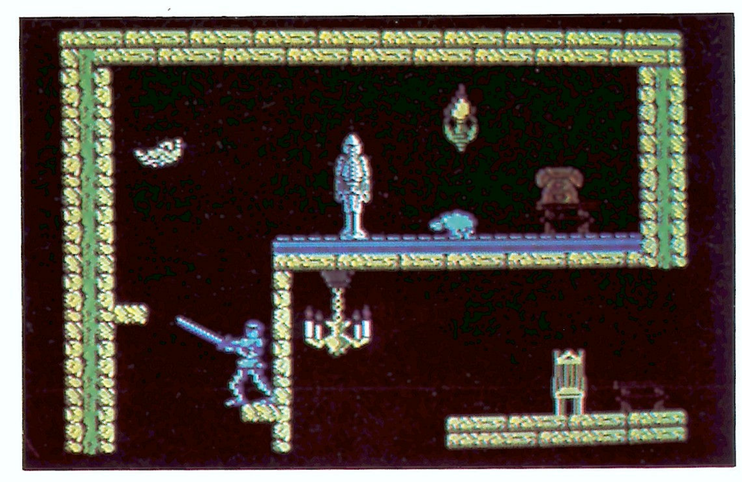
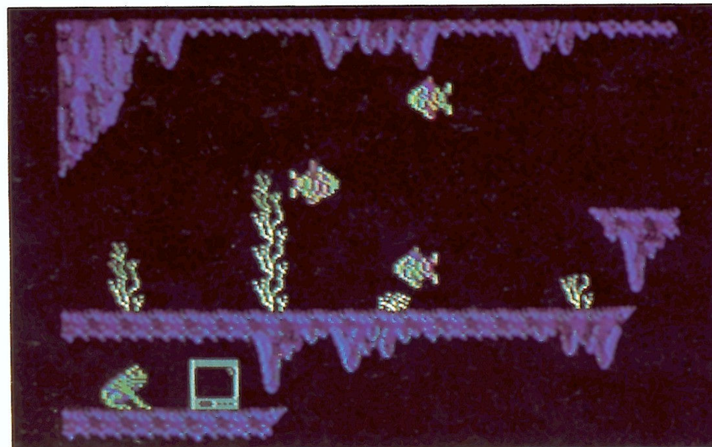
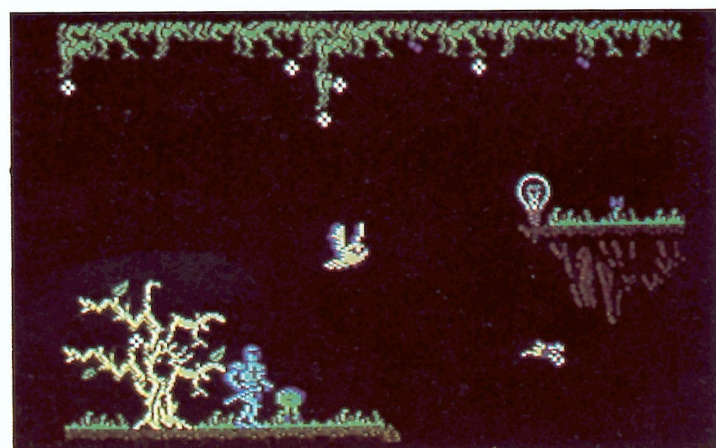
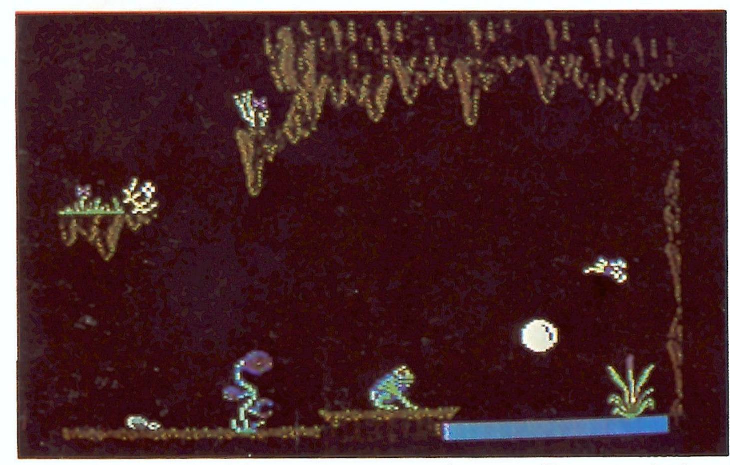
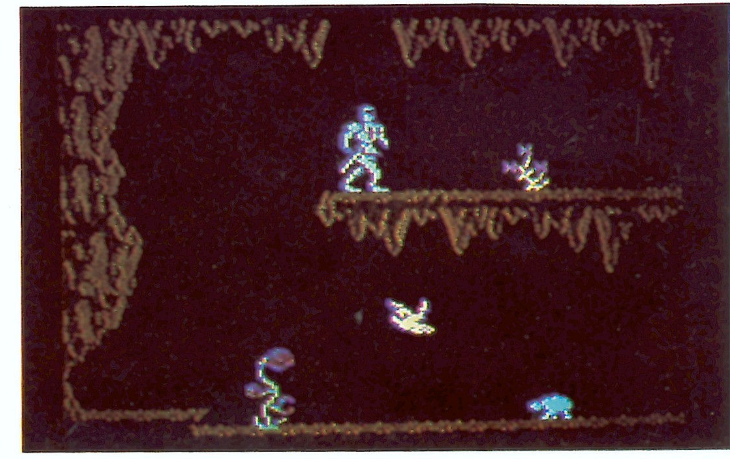
## Camelot Warriors. Realidad y ficción

La realidad de CAMELOT WARRIORS es, no obstante, mucho menos llamativa que la ficción, puesto que excluye todo el riesgo y todo el carácter épico de la aventura. CAMELOT WARRIORS tan solo es un paquete de software, fruto de la imaginación de un programador de ordenadores.

El juego, traducido íntegramente al castellano, tanto sobre la pantalla como en las instrucciones, se controla mediante un joystick o a través del teclado, con las teclas «Q», «O», «P» y la barra espaciadora. Curiosamente, éste es uno de los escasos programas donde resulta aconsejable no utilizar el joystick para así efectuar unos movimientos más precisos y exactos.

Tras superar la pantalla de presentación, gracias a la barra espaciadora o al dispa-



rador del joystick, deberás reaccionar deprisa, la aventura comienza ya, así que tal vez pierdas alguna que otra vida al principio. No te preocupes; dispones de un total de cinco vidas para salvar todos los impedimentos antepuestos entre tu persona y el cumplimiento de tu objetivo.

Podrás evitar estos obstáculos con facilidad si utilizas un poco tu capacidad de observación. De este modo, muy pronto apreciarás que los movimientos de tus enemigos son constantes, es decir, que nunca interviene en el juego ningún factor aleatorio. Teniendo presente este hecho, siguiendo las directrices generales suministradas por este artículo y con algo de habilidad, llegarás rápidamente a cumplir tu misión.

Después de leer los párrafos anteriores habréis notado que la habitual distensión de estas páginas ha desaparecido. Así se ha hecho por deseo expreso de nuestros redactores de mesa, quienes han visto en CAMELOT WARRIORS un fabuloso programa, por desgracia malogrado.

Malogrado porque apenas si algún sonido acompaña de vez en cuando el devenir de la aventura. Malogrado porque los gráficos, muy loables en su género simple y definido, necesitan una elaboración más exhaustiva, unos pequeños pero imprescindibles retoques. Malogrado porque la animación no es perfecta, ni mucho menos, sino saturada de errores.

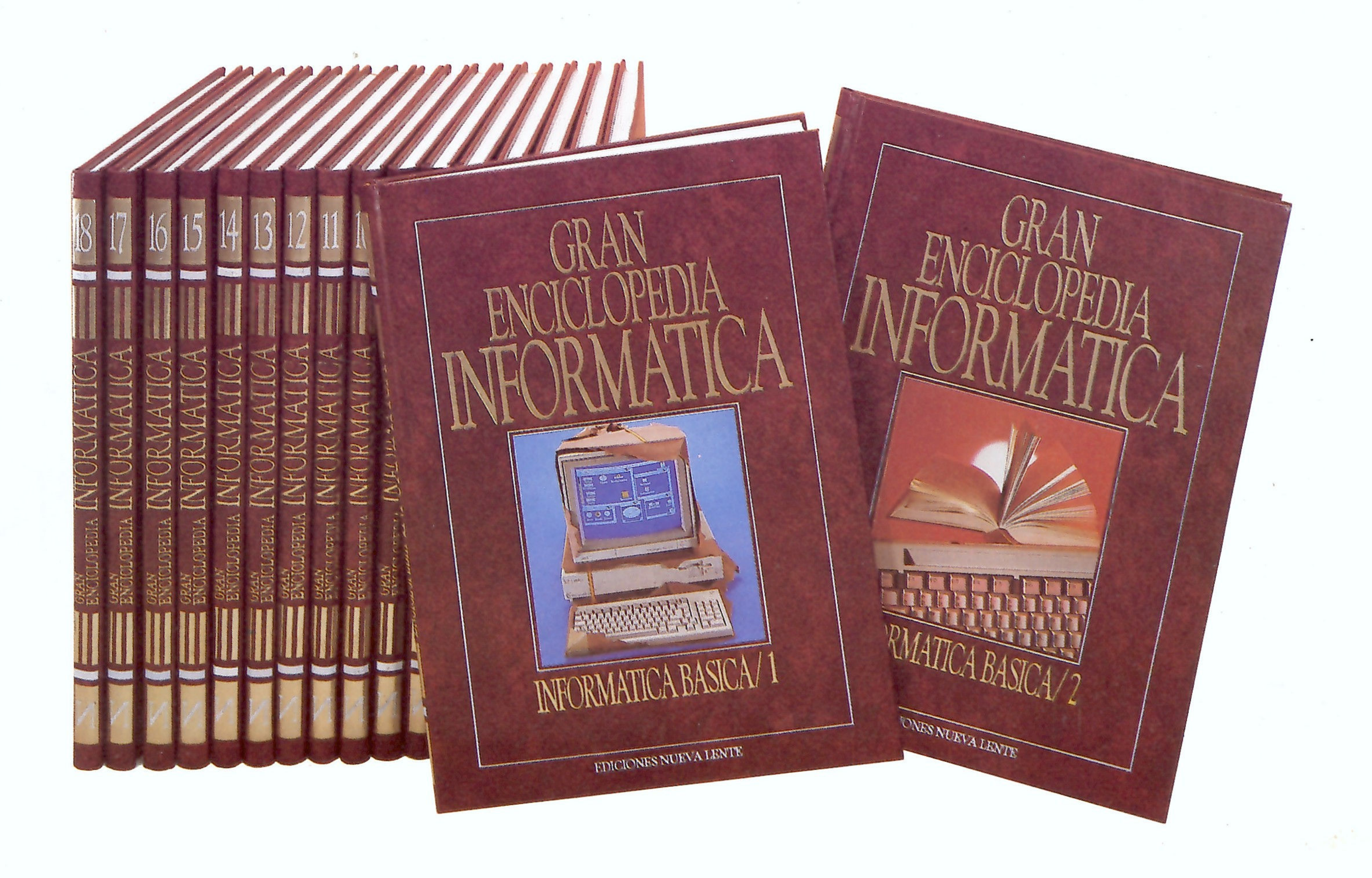
Malogrado porque no ofrece la más mínima variedad entre partida y partida. Malogrado porque ni siquiera aporta un cierto incremento de dificultad una vez superada la misión. Malogrado, en fin, porque el argumento se prestaba a mucho y el resultado se ha quedado en poco.

El resultado: un programa como tantos, de esos que, al cabo de unos días, se archivan en un rincón de la biblioteca de software y allí permanecen para siempre. Divertido en un principio e insoportable al final, a menos que decidas jugar a ciegas. Por ejemplo, cierra los ojos en la pantalla de presentación, pulsa la barra espaciadora y mantén pulsada la «Q» hasta que oigas catorce sonidos.

Abre los ojos...

**Toda la Informática a su alcance con**

# GRAN ENCICLOPEDIA DE LA INFORMATICA

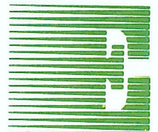



Un panorama total sobre Hardware, Software, Sistemas, Lenguajes en 18 tomos quincenales.

**¡YA ESTA A LA VENTA!**

# Superbase

SUPERBASE es uno de los mejores exponentes de lo que puede llegar a hacerse, dentro de la gestión integral de los datos, con un microordenador COMMODORE.

l programa dispone de versiones para COMMODORE 64 y COMMODORE 128, de la potente Base de Datos diseñada por Precisión Software y comercializada en España por Casa de Software.

SUPERBASE permite trabajar bajo dos sistemas. Uno «tutorial» guiado por Menús, y otro más directo dedicado a usuarios experimentados. Además, es capaz de admitir comandos tanto en modo directo como en modo programa: es decir, nos permite escribir una aplicación entera de gestión de la base de datos, en un «metalenguaje» bastante similar al de DBASE de Ashton Tate, escrito para el sistema operativo DOS de ordenadores IBM y compatibles, lo cual nos da idea de su potencia.

El equipo necesario para la instalación del programa es el siguiente: ordenador COMMODORE 64 o 128, una o dos unidades de disco 1541, impresora MPS 801 o equivalente, y receptor de TV o monitor.

Una vez introducido el disco que contiene el programa en su unidad (drive ø si trabajamos con dos unidades), debe escribirse: **LOAD "sb*", 8,1.** A continuación, se recibe el mensaje de extraer el disco de SUPERBASE e insertar el que alberga la base de datos.

De ser necesario formatear un disco virgen, podemos pulsar **F1** en este punto. Con ello, se nos solicita ubicar el disco a formatear en la unidad 0 y, después de dar la conformidad al mensaje de aviso, el nombre e identificador del disco.

## Copia del disco de Superbase

Dado que el disco de SUPERBASE se en-




cuentra protegido contra grabación, es necesario, después del formateo de cada diskete, la creación de una nueva base de datos. En este proceso, el programa va pidiendo alternativamente que introduzcamos el disco de SUPERBASE y el de datos, para permitir que se realicen las copias de las diferentes pantallas de ayuda que ilustran el programa.

## *El menú principal*

El Menú Principal consta de 8 opciones, que se corresponden con la pulsación de las diferentes teclas de función.

**F1 ENTER:** Tiene por objeto facilitar la entrada de datos a los ficheros.

**F2 SELECT:** Localiza y visualiza cualquier registro almacenado en SUPERBASE. Posee un menú propio, desde el cual puede escogerse una amplia gama de posibilidades.

**F3 FIND:** Permite localizar registros que respondan a un tipo determinado de criterios. Así mismo, almacena una lista de las claves de los registros seleccionados, para utilizarla en combinación con opciones del tipo **SORT** o **OUTPUT**.

**F4 OUTPUT:** Visualiza o imprime la información de todos los registros, o de una selección previa de éstos. Esta salida puede contener tanto textos como el contenido de campos, incluyendo variables BASIC y resultados de cálculos.

**F5 CALC:** Permite evaluar y/o visualizar cualquier expresión. Puede emplearse el rango completo de funciones BASIC, incluyendo las trigonométricas. Los resultados pueden almacenarse en campos, variables BASIC, o visualizarlos simplemente.

**F6 REPORT:** Da acceso a un grupo completo de comandos para generar informes impresos, a partir de la información base contenida en los ficheros de datos.

**F7 EXECUTE:** Permite pasar programas previamente confeccionados, los cuales pueden realizar automáticamente secuencias completas de operaciones con los ficheros. De esta forma, pueden ejecutarse, con la sola pulsación de una tecla, una complicada serie de comandos.

**F8 HELP:** Esta opción visualiza una pantalla de ayuda, conteniendo información sobre las facilidades proporcionadas por SUPERBASE.

## *El menú secundario*

Se accede a este menú por pulsación de **RETURN,** dentro del menú principal. El menú secundario consta de 8 opciones que se corresponden también con la pulsación de las diferentes teclas de función.

**F1 FILE:** Esta opción se emplea para cambiar de fichero durante el trabajo, o para crear otro fichero integrado en la base de datos, (pueden mantenerse hasta un máximo de 15).

**F2 FORMAT:** Tiene como cometido definir el formato de la pantalla de un fichero nuevo, o la de revisar el formato de la pantalla de uno existente. Cada fichero puede contener hasta cuatro de estas palabras.

**F3 BATCH:** Esta opción realiza los cálculos, utilizando la información de cada uno de los registros, o sólo de los seleccionados en el fichero actual. Es ésta, también, la opción empleada para realizar operaciones de actualización en los ficheros.

**F4 SORT:** Se utiliza la opción **SORT** para distribuir todos los registros, o una lista de ellos seleccionados por campos (no por el campo de clave). El resultado de dicha operación es un fichero, con las claves de los registros en el orden de los parámetros especificados. Con este procedimiento, el fichero original se mantiene sin modificaciones. La lista de registros producida en este apartado, pueden emplearse con **OUTPUT, BATCH, REPORT** o **SELECT**.

**F5 PROG:** Esta opción permite crear y almacenar en disco programas completos, descriptivos de las operaciones de la base de datos. **PROG** dota a SUPERBASE de la potencia de un Generador de Aplicaciones, proporcionando un BASIC extendido por todos los comandos de SUPERBASE.

**F6 MAINTAIN:** A través de esta opción se tiene acceso a un nuevo submenú de opciones para permitir realizar varias funciones de utilidad en los ficheros, incluyendo las funciones de **EXPORT** o **IMPORT** desde y para otros programas.

**F7 MEMO:** Permite crear pantallas de información que pueden ser llamadas a la hora de explotar la base de datos. De forma más concreta, pueden confeccionarse o modificar las pantallas de ayuda ya existentes en el programa, dependiendo de las necesidades concretas del usuario.

**F8 HELP:** Esta opción visualiza una pantalla de ayuda, de igual forma que puede obtenerse por pulsación de esta misma tecla dentro del menú principal.

## *Comandos en modo directo*

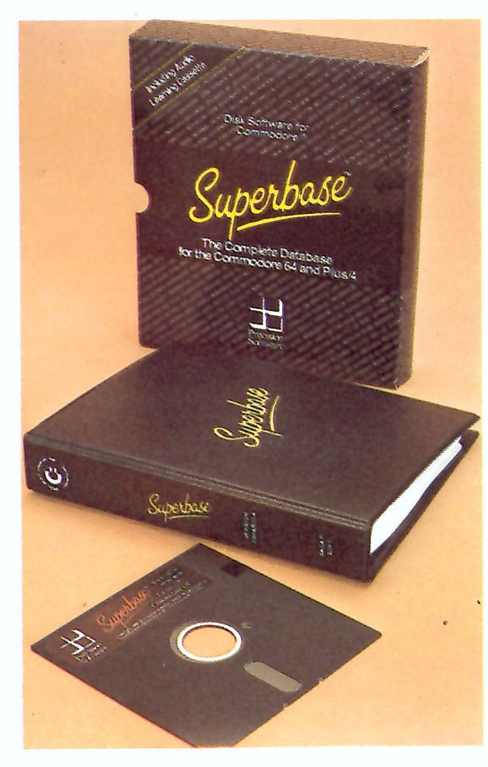
Aunque el programa está diseñado para permitir al usuario controlar sus muchas opciones mediante los dos menús explicados, también es posible abandonar el sistema de control por menú e introducir los comandos SUPERBASE directamente.

De esta forma, cada uno de los comandos incluidos en los dos menús anteriormente descritos, pueden escribirse en el área de comandos, junto con otros complementa-

rios proporcionados para el usuario avanzado. Por otro lado, pueden ejecutarse varios comandos en una misma línea separados por dos puntos «:».

Como norma general, los comandos deben incluirse en minúsculas. Sin embargo, y de forma similar a como pueden escribirse las palabras BASIC en forma abreviada, es posible nombrar las denominaciones de comando SUPERBASE, a través de la inicial y la segunda letra en mayúscula.

Siempre que se ejecuta una línea de comandos de esta manera, puede ser recuperada sin escribirla de nuevo, pulsando la tecla de flecha a la izquierda «—», situada en la esquina superior izquierda del teclado. Con ello, se visualiza en la linea de comandos lo último tecleado, para permitir una nueva ejecución o la modificación previa a la misma.




## *Formato de la entrada de datos*

SUPERBASE almacena información en ficheros de datos, donde la descripción de todos y cada uno de los campos que componen los registros han sido descritos por nosotros. Para ello, partimos de la pantalla del ordenador. Sobre ella vamos escribiendo los rótulos de petición de cada una de las informaciones. A continuación, reservamos el espacio que deba asignársele como máximo al campo de datos concreto.

Por otro lado, es necesario que especifiquemos la entidad de cada una de las clases de datos. Para el programa los datos pueden ser: Texto, Numéricos, Clave, Forzados, Fechas, Constantes, Resultados o calendario.

Como posibilidades opcionales, y de cara a mejorar la presentación en pantalla; es posible alterar los colores básicos del fondo, segundo plano y primero de la pantalla. Si lo hacemos, esta especificación de color pasa a formar parte del formato de petición de datos del fichero, y es respetado en posteriores apariciones en la pantalla.

Los campos de Texto almacenan contenidos alfanuméricos convencionales. Los Numéricos, contenidos de este tipo. Los de Clave, son similares a los de texto, pudiendo contener un máximo de 30 caracteres y sirviendo de campo de ordenamiento del fichero.

En los campos forzados, es necesario incorporar necesariamente información. Los de Fechas, albergan obligatoriamente 7 caracteres; dos para el día, tres para el mes (ENE, FEB...) y dos más para el año. En el caso de especificarse una longitud de 11 caracteres, es posible visualizar también el día de la semana.

Los campos constantes repiten información común a todos los registros, evitando tener que introducirla cada vez. Los de Resultado, permiten contener fórmulas que relacionan a otros campos del registro mediante fórmulas matemáticas, construidas a partir de los operadores BASIC conocidos.

Por último, los campos calendario, podemos considerarlos como campos de resultado pero referidos a operaciones con fechas. Pueden albergar por lo tanto, fechas de vencimiento a partir de un número de dias de una dada, etc...

## *Compatibilidad*

Otra de las habilidades de SUPERBASE es la de ser compatible con procesadores de texto como EASY SCRIPT, permitiendo la generación de aplicaciones conjuntas como el envio de cartas personalizadas («mailing»).

*Programa:* Superbase
*Distribuidor:* Casa de Software (sacati)
Taquígrafo Serra, 7
Tel.: (93) 321 96 36
08029 Barcelona
*Precio:* 22.500 ptas. (C-64)
y 26.500 ptas. (C-128).

# :RITEMAN: news

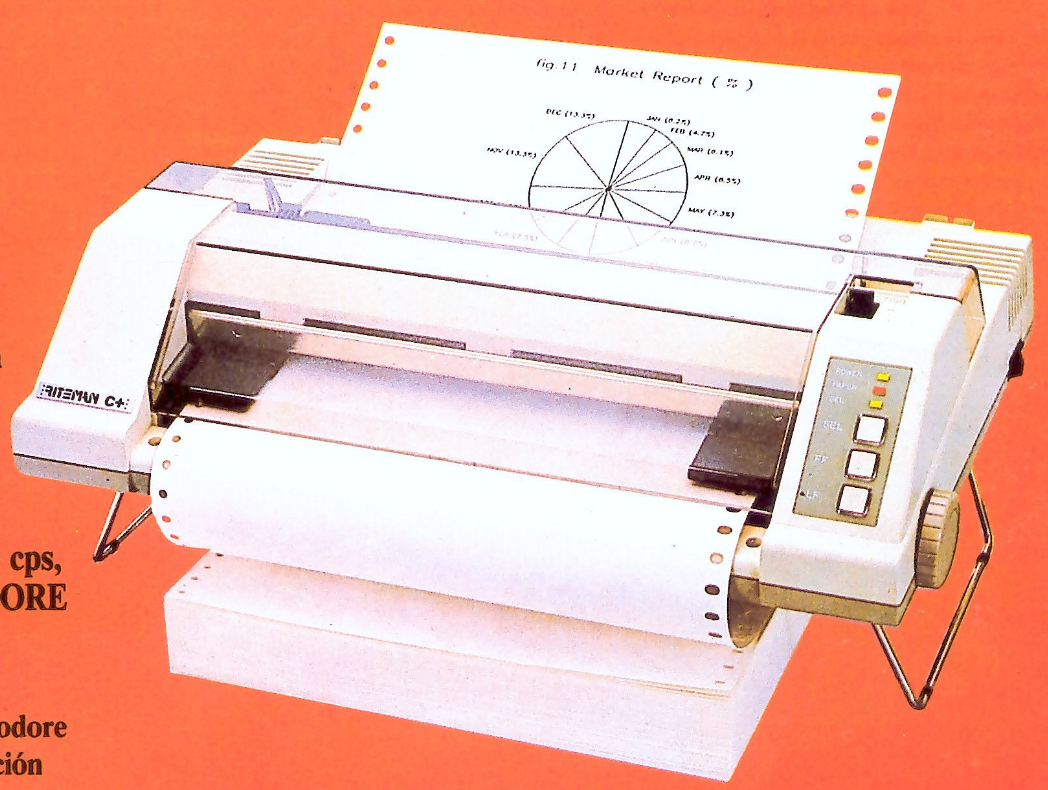
DATAMON REPRESENTACION EN ESPAÑA DE: :RITEMAN:

PROVENZA, 385-387
TEL. (93) 207 24 99*

TELEX 97791
08025 BARCELONA

## IMPRESORA PARA SU COMMODORE
### (óptima relación precio/prestaciones)

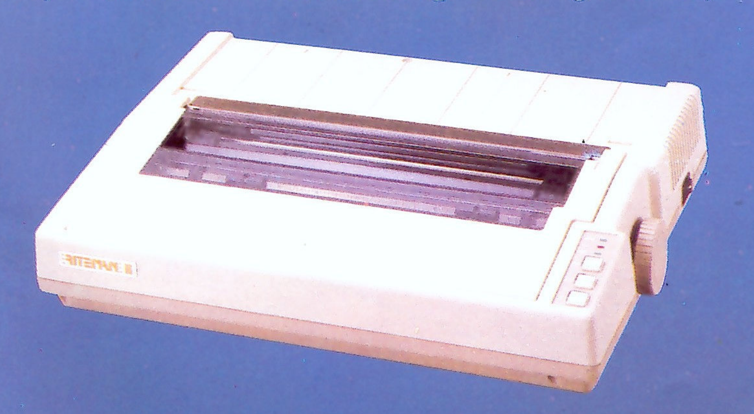
—Cabezal 9 agujas
—Doble operatividad
—Cinta autoretintada
—Tampón retintable
—Ausencia de rodillo
—No dobla el papel
—Elevadores inferiores
—Admite texto rígido
—Máximos tipos de escritura




**Modelo SUPER C+, 120 cps, NLQ, ASCII y COMMODORE**

• **Conexión directa a Commodore (cable incl.) Tracción y fricción**

## LA IMPRESORA PARA COMMODORE, ASCII Y PC'S COMPATIBLES
### (Máxima versatilidad/precio ajustado)



## RITEMAN 10-C

—140 cps, tracción y fricción
—Paralelo centronics/Commodore serie DIN
—Tablas ASCII y PC en Rom interna
—Tabla 100% Commodore y 8K RAM en módulo
—Interface Commodore exterior incluido
—RS 232-C opcional

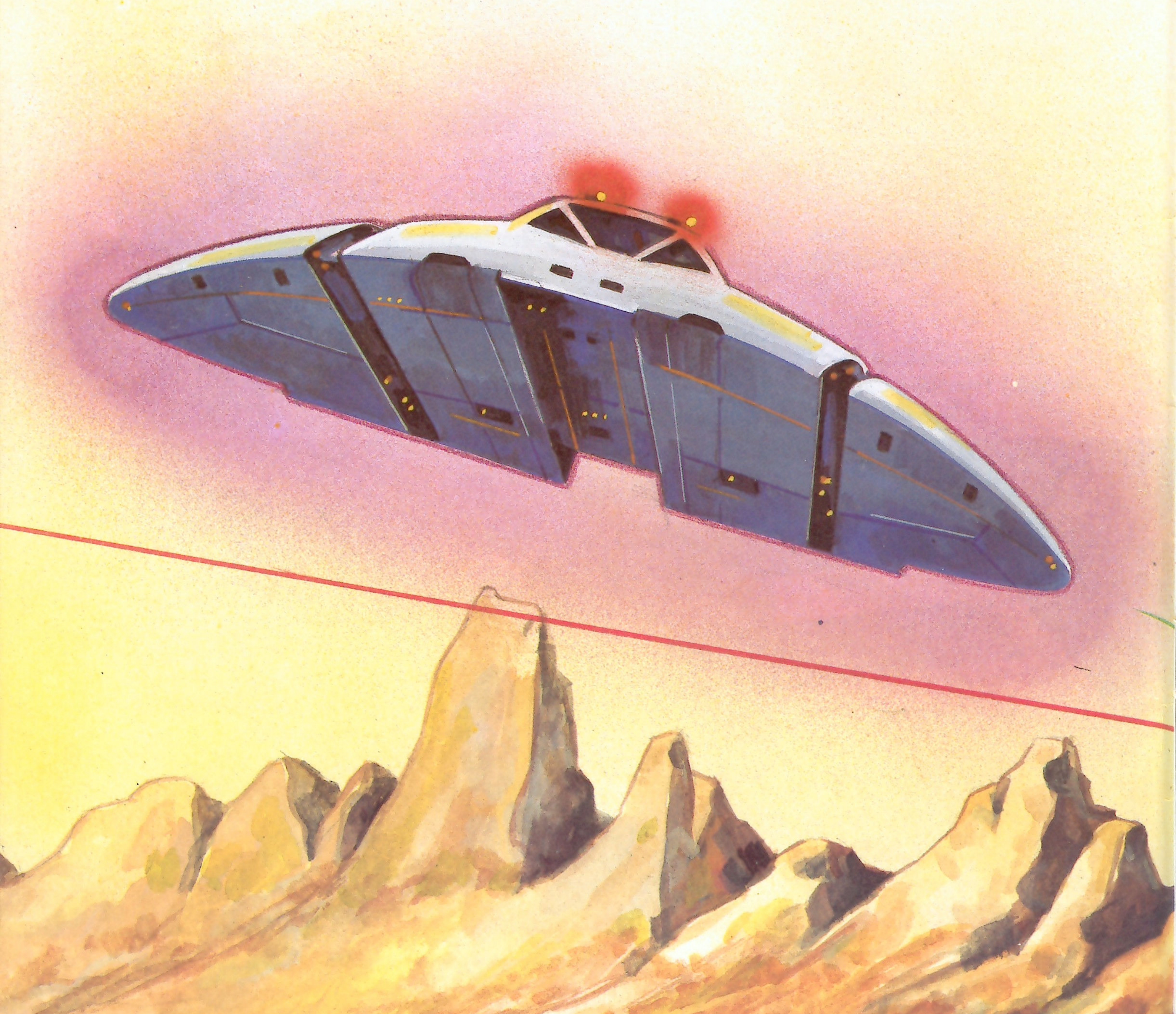
NOTA: Para Aplicaciones en las que se necesite más velocidad, o mayor tamaño de carro, también pueden aplicarse nuestros interfaces externos a los modelos RITEMAN 10/II y RITEMAN 15.

# Asteroides

Luke, Leia, Chewbaka, preparaos! La cuenta atrás comienza... ya. Tres... dos... uno... De pronto, todas las estrellas de la pantalla de visión se transforman en largas líneas luminosas y tanto Han Solo como el resto de los tripulantes del Halcón Milenario notan la presión sobre sus cuerpos. La velocidad de la espacionave triplica la de la luz: han entrado en el hiperespacio.

Los destellos de láser surcan el espacio por todas partes, mientras los tres cruceros estelares del imperio que te siguen se acercan cada vez más, ya casi rozan el timón de cola. No queda tiempo para calcular el salto al hiperespacio, pero es preciso huir: es la única forma de evitar una muerte segura.

Lógicamente, sus perseguidores no pueden saltar aún al hiperespacio, pues tal maniobra requiere más tiempo cuanto mayor es el tamaño de la astronave. Y los cruceros estelares son gigantescos, mucho más grandes que el resto de las naves de la Ga-



laxia. «¡Maldición, han escapado» —piensa furioso Darth Vader, comandante en jefe de las fuerzas oscuras.

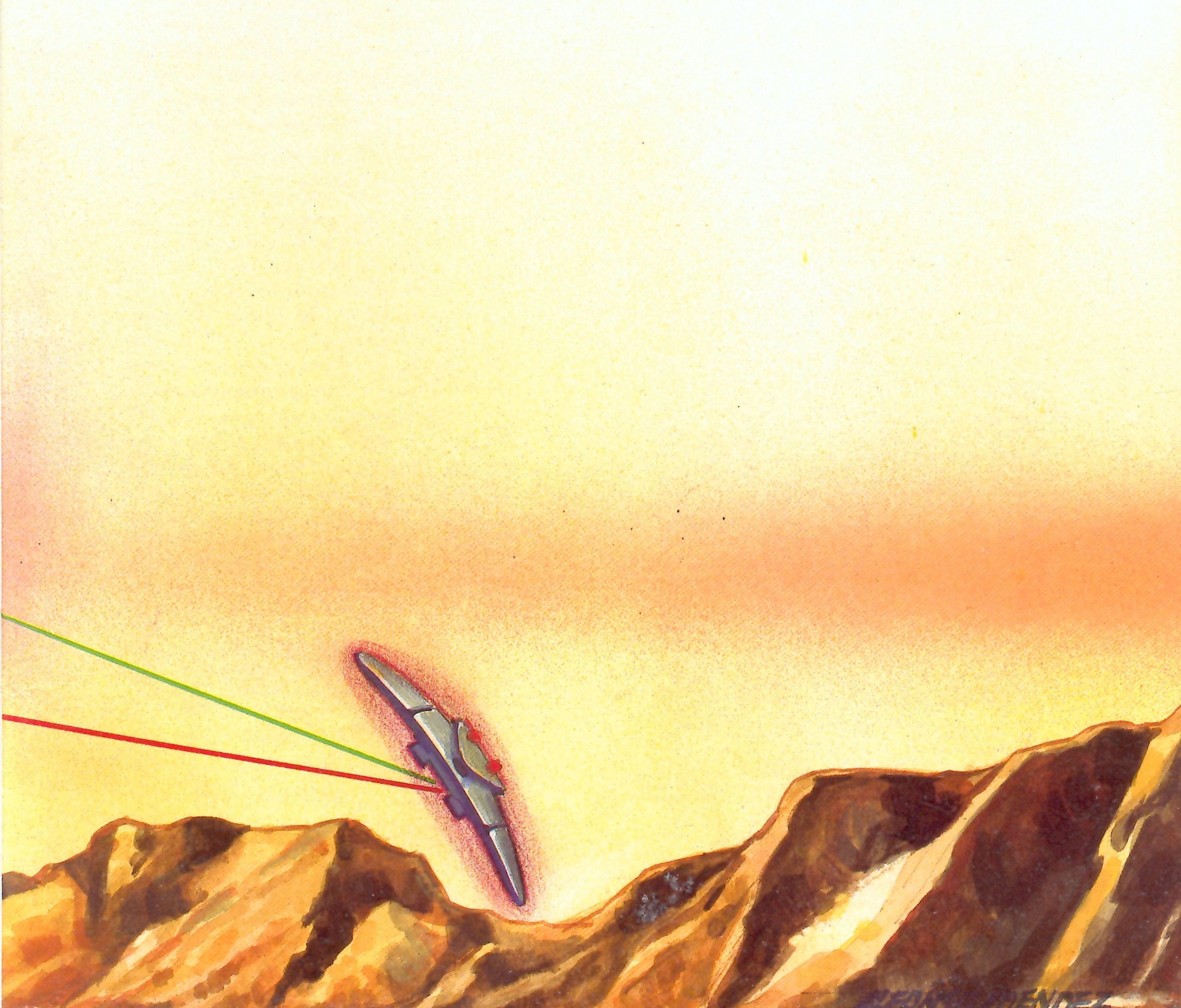
En efecto, han escapado de una muerte cierta, pero no están a salvo, ni mucho menos. El salto al hiperespacio les ha conducido hasta una zona perdida, surcada por grandes asteroides. Unos asteroides muy particulares, dado que están compuestos de magnetita, un mineral que siente atracción hacia casi todos los metales.

Esta magnetita dirige a todos los asteroides de la zona hacia el Halcón Milenario, el cual dispone de una única defensa: la artillería de a bordo. «¡Luke! al cañón; Chewbaka y yo dirigiremos la nave. Los demás agarraos bien. Hace falta mucha suerte para salir de esta situación...», grita Han Solo.

Todos sin discusión acatan las órdenes. Luke se encarga del control RETURN para lanzar proyectiles contra los asteroides. Al mismo tiempo, Chewbaca maneja con el SHIFT el propulsor de la nave. Han se encarga de utilizar los controles «S» y «D» para girar con la nave hacia la izquierda y hacia la derecha. Y por si las cosas se ponen feas, Leia permanece alerta junto al control «H», preparada para un nuevo salto hacia el Hiperespacio.

Luke confía en la fuerza y en la dureza del casco de la espacionave, que, según el informe técnico de R2-D2, es capaz de aguantar hasta tres impactos. De todas formas, si logran destruir el número necesario de asteroides para construir un escudo de magnetita, la nave aguantará más.

Esto marcha. Ya casi no quedan asteroides cuando de repente otra amenaza surge de las entrañas del hiperespacio: una nave alienígena que dispara contra el Halcón Milenario. Tal vez Han Solo y sus amigos logren sobrevivir, quizás... Pero lo más probable es que no lo consigan y deban inscribir su nombre para la posteridad.



# LISTADO

```
5 POKE 53280,0:POKE 53265,11:POKE 646,1:PRINTCHR$(8);
CHR$(142)                                            -025-
10 FOR R=0 TO 639:READ S:POKE 15360+R,S:NEXT R
                                                     -004-
20 FOR R=0 TO 111:READ S:POKE 14848+R,S:NEXT R
                                                     -000-
30 FOR R=0 TO 48:READ S:POKE 49152+R,S:NEXT R
                                                     -214-
40 FOR R=0 TO 421:READ S:POKE 49201+R,S:NEXT R
                                                     -253-
50 FOR R=0 TO 249:READ S:POKE 50000+R,S:NEXT R
                                                     -251-
100 SYS 49152:POKE 53272,30:POKE 53281,0             -216-
110 POKE 53287,1:POKE 53288,1                        -087-
120 FOR L=54272 TO 54295:POKE L,0:NEXT L             -073-
130 POKE 54296,15:POKE 54277,0:POKE 54278,240
                                                     -226-
140 POKE 49202,31:POKE 49454,69:POKE 49473,67:POKE 49
582,5                                                -230-
170 FOR R=0 TO 5:POKE 2042+R,240:POKE 53289+R,1:NEXT
R                                                    -127-
180 POKE 53253,50:POKE 53255,50:POKE 53257,50
                                                     -217-
190 POKE 53252,95:POKE 53254,115:POKE 53256,135
                                                     -070-
200 POKE 53259,50:POKE 53261,50:POKE 53263,50
                                                     -210-
210 POKE 53258,208:POKE 53260,228:POKE 53262,248
                                                     -117-
220 POKE 49211,1:POKE 49213,10                       -122-
230 POKE 49275,29:POKE 789,192                       -159-
240 POKE 54284,0:POKE 54285,240                      -187-
250 POKE 50017,254:RS=0                              -022-
1000 POKE 53269,0:POKE 53265,27:POKE 198,0:V1=3:V2=3:
A(1)=4:A(2)=4:T=1                                    -123-
1005 POKE 53278,0:S(1)=0:S(2)=0                      -147-
1010 PRINT"{CLR}{11 ABJ}"TAB(6)"UNO (1) O DOS (2) JUG
ADORES"                                              -019-
1020 FOR X=1 TO 50:GET W$:J=VAL(W$):IF J=1 OR J=2 THE
N 1060                                               -142-
1025 NEXT X                                          -159-
1030 PRINTTAB(6)"{ARB}[29 ESP]"                      -068-
1040 FOR X=1 TO 50:GET W$:J=VAL(W$):IF J=1 OR J=2 THE
N 1060                                               -144-
1045 NEXT X                                          -161-
1050 GOTO 1010                                       -001-
1051 POKE 53001,1:POKE 53000,4:POKE 53002,0:POKE 5300
3,0                                                  -044-
1052 T=PEEK(53001):A(1)=PEEK(53000):S(1)=PEEK(53002)*
100:S(2)=PEEK(53003)*100                             -128-
1055 V1=PEEK(53004):V2=PEEK(53005):J=PEEK(53006):A(2)
=PEEK(53007)                                         -120-
1060 POKE 53265,11:PRINT"{CLR}":GOSUB 1300           -005-
1065 D=0:XD=0:XY=0:DX=0:DY=0:NA=0                    -245-
1070 POKE 2040,240:POKE 53248,155:POKE  53249,137
                                                     -140-
1080 POKE 53269,255:POKE 53250,0:POKE 53251,0
                                                     -210-
1100 PRINT"{HOM}JUGADOR 1[8 ESP]RECORD[8 ESP]JUGADOR
2"                                                   -064-
1105 S1$="[8 ESP]"+STR$(S(1)):S1$=RIGHT$(S1$,6)
                                                     -154-
1107 S2$="[8 ESP]"+STR$(S(2)):S2$=RIGHT$(S2$,6)
                                                     -160-
1108 R$="[9 ESP]"+STR$(RS):R$=RIGHT$(R$,6)           -247-
1110 PRINT"{ARB}{DCH}";S1$,"{7 DCH}";R$,"{2 DCH}";S2$
                                                     -191-
1120 POKE 53250,0:POKE 53251,0:POKE 53264,0          -092-
1130 IF V1=4 THEN N=31                               -234-
1140 IF V1=3 THEN N=27                               -239-
1150 IF V1=2 THEN N=19                               -240-
1160 IF V1=1 OR V1=0 THEN N=3                        -142-
1170 IF J=1 THEN 1230                                -133-
1180 IF V2=4 THEN N1=224                             -085-
1190 IF V2=3 THEN N1=192                             -089-
1200 IF V2=2 THEN N1=128                             -079-
1210 IF V2=1 OR V2=0 THEN N1=0                       -186-
1220 N=N+N1                                          -104-
1230 POKE 53269,N                                    -184-
1240 POKE 10,0:POKE 251,0:POKE 252,0:POKE 253,0:POKE
254,0:POKE 2,0                                       -004-
1295 GOTO 2000                                       -012-
1300 FOR R=1 TO A(T)                                 -116-
1310 X=INT(RND(1)*11):Y=INT(RND(1)*7)+3              -217-
1320 T(R)=3                                          -077-
1330 IF X AND 1 THEN X(R)=X+24:GOTO 1340             -029-
1335 X(R)=X+1                                        -216-
1340 IF Y AND 1 THEN Y(R)=Y+12:GOTO 1350             -031-
1345 Y(R)=Y                                          -127-
1350 POKE 253,Y(R):POKE 254,X(R):SYS 50104           -152-
1360 U(R)=INT(RND(1)*3)-1:V(R)=INT(RND(1)*3)-1
                                                     -073-
1370 IF U(R)=0 AND V(R)=0 THEN 1360                  -177-
1400 NEXT R:RETURN                                   -176-
2000 POKE 53265,27                                   -203-
2010 POKE 2,0:POKE 10,0:POKE 53278,240               -090-
2020 FOR R=1 TO A(T)                                 -116-
2030 ON T(R) GOSUB 5000,5100,5200                    -003-
2035 IF D THEN GOSUB 7200                            -181-
2040 ON PEEK(2) GOSUB 6000,7000                      -196-
2050 IF PEEK(10) THEN 3000                           -159-
2060 NEXT R:GOTO 2020                                -240-
3000 POKE 53278,0:FOR N=0 TO 10                      -215-
3010 POKE 53287,0:FOR N1=1 TO 20:NEXT N1             -035-
3020 POKE 53287,1:FOR N1=1 TO 20:NEXT N1             -037-
3030 POKE 54283,128:POKE 54280,10:POKE 54279,30:POKE
54283,129                                            -149-
3040 NEXT N:POKE 54283,128                           -234-
3050 POKE 53269,PEEK(53269)AND254                    -089-
3060 IF J=2 THEN 3100                                -132-
3070 V1=V1-1:IF V1 THEN 1060                         -057-
3080 GOTO 10000                                      -053-
3100 IF T=2 THEN 3130                                -140-
3110 V1=V1-1:IF V1=0 AND V2=0 THEN 10000             -211-
3120 T=2:GOTO 1060                                   -003-
3130 V2=V2-1:IF V1=0 AND V2=0  THEN 10000            -247-
3140 T=1:GOTO 1060                                   -004-
5000 POKE 253,Y(R):POKE 254,X(R):SYS 50014           -148-
5005 X(R)=X(R)+U(R):IF X(R)=37 THEN X(R)=1:GOTO 5020
                                                     -253-
5010 IF X(R)=0 THEN X(R)=36                          -013-
5020 Y(R)=Y(R)+V(R):IF Y(R)=3 THEN Y(R)=22:GOTO 5040
                                                     -253-
5030 IF Y(R)=23 THEN Y(R)=4                          -017-
5040 POKE 253,Y(R):POKE 254,X(R):SYS 50000:RETURN
                                                     -173-
5100 POKE 253,Y(R):POKE 254,X(R):SYS 50066           -156-
5105 X(R)=X(R)+U(R):IF X(R)=36 THEN X(R)=1:GOTO 5120
                                                     -254-
5110 IF X(R)=0 THEN X(R)=35                          -013-
5120 Y(R)=Y(R)+V(R):IF Y(R)=3 THEN Y(R)=21:GOTO 5140
                                                     -254-
5130 IF Y(R)=22 THEN Y(R)=4                          -017-
5140 POKE 253,Y(R):POKE 254,X(R):SYS 50028:RETURN
                                                     -184-
5200 POKE 253,Y(R):POKE 254,X(R):SYS 50177           -160-
5205 X(R)=X(R)+U(R):IF X(R)=35 THEN X(R)=1:GOTO 5220
                                                     -255-
5210 IF X(R)=0 THEN X(R)=34                          -013-
5220 Y(R)=Y(R)+V(R):IF Y(R)=3 THEN Y(R)=20:GOTO 5240
                                                     -255-
5230 IF Y(R)=21 THEN Y(R)=4                          -017-
5240 POKE 253,Y(R):POKE 254,X(R):SYS 50104:RETURN
                                                     -180-
6000 POKE 54280,21:POKE 54279,27:POKE 54283,17
                                                     -019-
6005 X=INT(RND(1)*250)+50:IF X>255 THEN POKE 53264,PE
EK(53264)OR1:X=X-255                                 -184-
6010 POKE 53269,PEEK(53269)AND254:POKE 53264,PEEK(532
64)AND254                                            -249-
6020 X=INT(RND(1)*250)+50:IF X>255 THEN POKE 53264,PE
EK(53264)OR1:X=X-255                                 -181-
6030 Y=INT(RND(1)*120)+70                            -112-
6040 POKE 53248,INT(X/2)*2+1:POKE 53249,INT(Y/2)*2+1
                                                     -130-
6050 POKE 53269,PEEK(53269)OR1                       -192-
6060 POKE 54280,100:POKE 54279,181:POKE 54283,33
                                                     -118-
6070 POKE 54283,32:POKE 2,0:RETURN                   -004-
7000 POKE 2,0:IF D THEN RETURN                       -064-
7005 POKE 54280,230:POKE 54279,192:POKE  54283,17
                                                     -158-
7010 D=PEEK(2040)-239: ON D GOSUB 7100,7110,7120,7130
,7140,7150,7160,7170                                 -027-
7020 DX=INT(PEEK(53248)/8.5):IF PEEK(53264)=1 THEN DX
=DX+30                                               -135-
7030 DY=INT(PEEK(53249)/11.25):DY=DY+YD              -028-
7035 IF DY<7 THEN DY=DY-1                            -242-
7037 IF DY>14 THEN DY=DY+1                           -034-
7038 DX=DX+XD+XD:DY=DY+YD                            -110-
7039 IF DX<1 OR DX>37 OR DY<4 OR DY>22 THEN D=0:XD=0:
YD=0:DX=0:DY=0:D=0:RETURN                            -017-
7040 POKE 1024+(DY*40)+DX,65:POKE 54283,16:RETURN
                                                     -066-
7100 XD=0:YD=-1:RETURN                               -125-
7110 XD=1:YD=-1:RETURN                               -127-
7120 XD=1:YD=0:RETURN                                -082-
7130 XD=1:YD=1:RETURN                                -084-
7140 XD=0:YD=1:RETURN                                -084-
7150 XD=-1:YD=1:RETURN                               -131-
7160 XD=-1:YD=0:RETURN                               -131-
7170 XD=-1:YD=-1:RETURN                              -178-
7200 POKE 1024+(DY*40)+DX,32                         -254-
7210 DX=DX+XD:DY=DY+YD                               -159-
7220 N=1024+(DY*40)+DX:IF PEEK(N)<>32 THEN GOSUB 7500
:RETURN                                              -044-
7230 IF DX=0 OR DX=38 OR DY=3 OR DY=23 THEN D=0:XD=0:
YD=0:N=0:RETURN                                      -141-
7240 POKE N,65:RETURN                                -059-
7500 POKE 54284,0:POKE 54286,2                       -142-
7505 N=0:POKE 54280,5:POKE 54279,4:POKE 54283,129
                                                     -228-
7510 FOR Z=1 TO A(T)                                 -133-
7520 ON T(Z) GOTO 7700,7800,7900                     -232-
7530 NEXT Z:POKE 54286,240:RETURN                    -022-
7610 N=1:T(Z)=T(Z)-1:POKE 54286,240:POKE 54283,128:GO
SUB 9000:RETURN                                      -228-
7700 IF DX<>X(Z) OR DY<>Y(Z) THEN 7530               -240-
7710 POKE 253,Y(Z):POKE 254,X(Z):SYS 50014:GOTO 7610
                                                     -015-
7800 FOR Z1=0 TO 1:FOR Z2=0 TO 1                     -080-
7810 IF DX=X(Z)+Z2 AND DY=Y(Z)+Z1 THEN 7830          -026-
7820 NEXT Z2,Z1:GOTO 7530                            -245-
7830 POKE 253,Y(Z):POKE 254,X(Z):SYS 50066:GOTO 7610
                                                     -025-
7900 FOR Z1=0 TO 2:FOR Z2=0 TO 2                     -083-
7910 IF DX=X(Z)+Z2 AND DY=Y(Z)+Z1 THEN  7930
                                                     -060-
7920 NEXT Z2,Z1:GOTO 7530                            -246-
7930 POKE 253,Y(Z):POKE 254,X(Z):SYS 50177:GOTO 7610
                                                     -029-
9000 NA=NA+1:S(T)=S(T)+500:S$="[10 ESP]"+STR$(S(T))
                                                     -202-
9005 IF NA=3*A(T) THEN 9500                          -233-
9010 IF S(T)=10500 THEN 9100                         -253-
```

```
9050 IF T=1 THEN PRINT"{HOM}{ABJ}{DCH}"RIGHT$(S$,6):R
ETURN                                          -198-
9060 PRINT"{HOM}{ABJ}",,,RIGHT$(S$,8):RETURN
                                               -080-
9100 IF J=1 THEN V2=0                          -181-
9102 IF T=1 THEN V1=V1+1:GOTO 9110             -209-
9105 V2=V2+1                                   -152-
9110 IF V1=4 THEN N=31                         -240-
9115 IF V1=3 THEN N=27                         -249-
9120 IF V1=2 THEN N=19                         -245-
9130 IF V1=1 OR V1=0 THEN N=3                  -147-
9200 IF V2=4 THEN N1=224                       -086-
9210 IF V2=3 THEN N1=192                       -090-
9220 IF V2=2 THEN N1=128                       -089-
9230 IF V2=1 OR V2=0 THEN N1=0                 -196-
9240 N=N+N1:POKE 53269,N                       -126-
9250 FOR ZZ=1 TO 3                             -047-
9260 POKE 54280,100:POKE 54279,50:POKE 54283,33
                                               -070-
9265 FOR Z1=1 TO 10:NEXTZ1                     -062-
9270 POKE 54280,50:POKE 54279,100:POKE  54283,17
                                               -105-
9275 FOR Z1=1 TO 10:NEXT Z1                    -095-
9280 NEXT ZZ:POKE 54283,16:RETURN              -066-
9500 A(T)=A(T)+1                               -083-
9510 POKE 53000,A(1):POKE 53001,T:POKE 53002,S(1)/100
:POKE 53003,S(2)/100                           -177-
9515 POKE 53004,V1:POKE 53005,V2:POKE 53006,J:POKE 53
007,A(2)                                       -161-
9520 RUN 1052                                  -205-
10000 POKE 53269,0:POKE 53278,0                -179-
10010 FOR ZZ=1 TO 10                           -127-
10020 PRINT"{HOM}{11 ABJ}"TAB(15)"GAME OVER"
                                               -250-
10030 FOR Z1=1 TO 50:NEXT Z1                   -128-
10040 PRINTTAB(15)"{ARB}[14 ESP]"              -197-
10050 FOR Z1=1 TO 50:NEXT Z1                   -130-
10060 NEXT ZZ                                  -042-
10070 IF S(1)>RS THEN RS=S(1)                  -165-
10080 IF S(2)>RS THEN RS=S(2)                  -168-
10090 GOTO 1000                                -052-
40000 REM *** SPRITES ***                      -126-
40005 DATA 0,24,0,0,24,0,0,60,0,0,60,0,0,102,0,0
                                               -034-
40010 DATA 102,0,0,195,0,0,195,0,1,129,128,1,129,128,
3,24                                           -013-
40020 DATA 192,3,60,192,3,102,192,3,195,192,3,129,192
,3,0,192                                       -239-
40030 DATA 3,0,192,0,0,0,0,0,0,0,0,0,0,0,0
                                               -084-
40040 DATA 0,0,0,0,0,0,0,0,0,0,24,0,0,248,0
                                               -138-
40050 DATA 3,240,0,30,48,0,120,48,0,224,48,0,192,112,
3,252                                          -062-
40060 DATA 96,7,6,224,6,6,192,0,7,192,0,7,128,0,6,0
                                               -255-
40070 DATA 0,14,0,0,12,0,0,0,0,0,0,0,0,0,0
                                               -081-
40080 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,1       -235-
40090 DATA 254,0,1,255,128,0,49,240,0,24,124,0,12,15,
0,12                                           -010-
40100 DATA 15,0,24,124,0,49,240,1,255,128,1,254,0,0,0
,0                                             -157-
40110 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0       -228-
40120 DATA 0,0,0,0,0,0,0,0,0,12,0,0,14,0,0
                                               -077-
194 DATA 6,0,0,7,128,0,7,192,6,6,192,7,6,224,3,252
                                               -208-
40140 DATA 96,0,192,112,0,224,48,0,120,48,0,30,48,0,3
,240                                           -017-
40150 DATA 0,0,248,0,0,24,0,0,0,0,0,0,0,0,0
                                               -140-
40160 DATA 0,0,0,0,0,0,0,0,0,0,0,3,0,192,3
                                               -091-
40170 DATA 0,192,3,129,192,3,195,192,3,102,192,3,60,1
92,3,24                                        -191-
40180 DATA 192,1,129,128,1,129,128,0,195,0,0,195,0,0,
102,0                                          -072-
40190 DATA 0,102,0,0,60,0,0,60,0,0,24,0,0,24,0,0
                                               -039-
40200 DATA 0,0,0,0,0,0,0,0,0,48,0,0,112,0,0
                                               -132-
40210 DATA 96,0,1,224,0,3,224,0,3,96,96,7,96,224,6,63
                                               -102-
40220 DATA 192,14,3,0,12,7,0,12,30,0,12,120,0,15,192,
0                                              -098-
40230 DATA 31,0,0,24,0,0,0,0,0,0,0,0,0,0,0
                                               -081-
40240 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0       -232-
40250 DATA 0,0,0,127,128,1,255,128,15,140,0,62,24,0,2
40,48                                          -065-
40260 DATA 0,240,48,0,62,24,0,15,140,0,1,255,128,0,12
7,128                                          -066-
40270 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0       -235-
40280 DATA 0,0,0,0,0,0,0,0,24,0,0,31,0,0,15
                                               -140-
40290 DATA 192,0,12,120,0,12,30,0,12,7,0,14,3,0,6,63
                                               -006-
40300 DATA 192,7,96,224,3,96,96,3,224,0,1,224,0,0,96,
0                                              -147-
40310 DATA 0,112,0,0,48,0,0,0,0,0,0,0,0,0,0
                                               -134-
40320 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0       -231-
40330 DATA 0,0,0,30,0,0,115,128,0,192,192,3,255,240,0
,225                                           -003-
40340 DATA 192,0,63,0,0,0,0,0,0,0,0,0,0,0,0
                                               -142-
40350 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0       -234-
40360 DATA 0,0,0,0,0,0,0,0,0,0,0,28,0,0        -037-
40370 DATA 54,0,0,99,0,1,193,192,3,0,96,31,255,252,31
,255                                           -044-
40380 DATA 252,1,193,192,0,99,0,0,62,0,0,0,0,0,0
                                               -170-
40390 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0       -238-
41000 REM *** CARACTERES ***                   -050-
41005 DATA 120,102,195,195,102,198,102,60      -026-
41010 DATA 0,0,0,24,24,0,0,0                   -112-
41020 DATA 0,30,51,48,48,96,192,192            -253-
41030 DATA 192,240,28,6,6,7,0,0                -037-
41040 DATA 3,6,12,24,224,128,0,0               -076-
41050 DATA 0,0,240,24,30,14,3,3                -018-
41060 DATA 0,0,0,3,7,12,24,48                  -184-
41070 DATA 0,0,24,254,231,1,0,0                -018-
41080 DATA 0,0,0,0,0,128,192,96                -033-
41090 DATA 48,24,12,12,24,24,24,24             -188-
41100 DATA 48,24,24,24,24,24,48,48             -198-
41110 DATA 24,12,6,3,1,0,0,0                   -120-
41120 DATA 0,7,15,24,240,96,0,0                -030-
41130 DATA 96,192,128,0,0,0,0,0                -029-
42000 REM *** CARGADOR/ LM ***                 -129-
42005 DATA 173,14,220,41,254,141,14,220,165,1,41,251,
133,1,162,0                                    -103-
42010 DATA 189,0,208,157,0,56,232,208,247,189,0,209,1
57,0,57,232                                    -148-
42020 DATA 208,247,165,1,9,4,133,1,173,14,220,9,1,141
,14,220                                        -172-
42030 DATA 96                                  -194-
43000 REM *** LM/ CONTROL NAVE ***             -170-
43005 DATA 173,30,208,41,1,201,1,208,7,169,3,133,2,76
,75,192                                        -190-
43010 DATA 41,2,201,2,208,4,169,4,133,2,166,251,224,8
,240,6                                         -126-
43020 DATA 232,134,251,76,187,192,162,0,134,251,173,1
97,0,201,13,208                                -077-
43030 DATA 3,76,143,192,201,18,208,3,76,165,192,201,6
0,208,3,76                                     -093-
43040 DATA 187,192,201,1,208,3,76,136,192,201,60,240,
2,76,49,234                                    -141-
43050 DATA 169,1,133,2,76,49,234,169,2,133,2,76,49,23
4,174,248                                      -063-
43060 DATA 7,224,240,208,8,162,247,142,248,7,76,49,23
4,202,142,248                                  -000-
43070 DATA 7,76,49,234,174,248,7,224,247,208,8,162,24
0,142,248,7                                    -169-
43080 DATA 76,49,234,232,142,248,7,76,49,234,76,166,1
93,234,234,234                                 -072-
43090 DATA 234,234,234,234,234,234,234,234,234,173,24
8,7,201,240,208,6                              -184-
43100 DATA 32,42,193,76,34,193,201,241,208,9,32,42,19
3,32,72,193                                    -143-
43110 DATA 76,34,193,201,242,208,6,32,72,193,76,34,19
3,201,243,208                                  -245-
43120 DATA 9,32,57,193,32,72,193,76,34,193,201,244,20
8,6,32,57                                      -057-
43130 DATA 193,76,34,193,201,245,208,9,32,57,193,32,1
19,193,76,34                                   -215-
43140 DATA 193,201,246,208,6,32,119,193,76,34,193,32,
42,193,32,119                                  -252-
43150 DATA 193,234,234,234,234,234,76,49,234,174,1,20
8,224,39,208,2                                 -047-
43160 DATA 162,239,202,202,142,1,208,96,174,1,208,224
,237,208,2,162                                 -028-
43170 DATA 37,232,232,142,1,208,96,174,0,208,173,16,2
08,41,1,201                                    -129-
43180 DATA 1,208,20,224,75,208,10,173,16,208,41,2,141
,16,208,162                                    -125-
43190 DATA 255,232,232,142,0,208,96,224,255,208,246,1
73,16,208,9,1                                  -249-
43200 DATA 141,16,208,76,98,193,174,0,208,173,16,208,
41,1,201,1                                     -085-
43210 DATA 208,18,224,1,208,8,173,16,208,41,2,141,16,
208,202,202                                    -120-
43220 DATA 142,0,208,96,224,11,208,246,173,16,208,9,1
,141,16,208                                    -134-
43230 DATA 162,77,76,143,193,173,141,2,201,1,208,19,1
62,10,134,252                                  -233-
43240 DATA 169,125,141,0,212,169,7,141,1,212,169,129,
141,4,212,166                                  -233-
43250 DATA 252,224,0,240,6,202,134,252,76,202,192,169
,128,141,4,212                                 -021-
43260 DATA 76,49,234,76,202,192                -069-
44000 REM *** ASTER/ LM ***                    -191-
44005 DATA 166,253,164,254,24,32,240,255,169,96,32,21
0,255,96,166,253                               -158-
44010 DATA 164,253,24,32,240,255,169,32,32,210,255,96
,166,253,164,254                               -142-
44020 DATA 24,32,240,255,169,98,32,210,255,169,101,32
,210,255,166,253                               -135-
44030 DATA 164,254,232,24,32,240,255,169,99,32,210,25
5,169,100,32,210                               -130-
44040 DATA 255,96,166,253,164,254,24,32,240,255,169,3
2,32,210,255,169                               -152-
44050 DATA 32,32,210,255,166,253,164,254,232,24,32,24
0,255,169,32,32                                -079-
44060 DATA 210,255,169,32,32,210,255,96,166,253,164,2
54,24,32,240,255                               -141-
44070 DATA 169,102,32,210,255,169,103,32,210,255,169,
104,32,210,255,166                             -226-
44080 DATA 253,164,254,232,24,32,240,255,169,105,32,2
10,255,169,32,32                               -134-
44090 DATA 210,255,169,106,32,210,255,166,253,164,254
,232,232,24,32,240                             -229-
44100 DATA 255,169,107,32,210,255,169,108,32,210,255,
169,109,32,210,255                             -234-
44110 DATA 96,166,253,164,254,24,32,240,255,169,32,32
,210,255,169,32                                -095-
44120 DATA 32,210,255,169,32,32,210,255,166,253,164,2
54,232,24,32,240                               -123-
44130 DATA 255,169,32,32,210,255,169,32,32,210,255,16
9,32,32,210,255                                -081-
44140 DATA 166,253,164,254,232,232,24,32,240,255,169,
32,32,210,255,169                              -188-
44150 DATA 32,32,210,255,169,32,32,210,255,96
                                               -229-

READY.                                         -173-
```




***TU MICRO COMMODORE ha creado el revolucionario sistema de introducción de programas FUERA ERRORES. Este nos permitirá introducir, sin temor alguno al esfuerzo inútil, cualquier listado por largo y complicado que parezca.***

Para adoptar los listados publicados bajo este sistema, deberemos seguir las siguientes normas:

1) Es fundamental transcribir EXACTAMENTE el listado reproducido, incluyendo todos sus espacios, aunque se trate de separaciones entre número de instrucción y línea de instrucción.

2) Todas las líneas finalizarán con un número de tres dígitos, encerrado entre guiones, que NO deberá ser introducido, puesto que no forma parte del programa, sino que tiene la finalidad de hacer funcionar el sistema FUERA ERRORES, según veremos más adelante. Para evitar equivocaciones, dicha cifra entre guiones se sitúa en el margen derecho del final de la línea BASIC a la cual corresponde, a una distancia prudencial del mismo.

3) Para facilitar la introducción de símbolos difícilmente interpretables, se procede a la siguiente representación en los listados.

– Las letras aparecidas entre menor y mayor deberán ser introducidas con pulsación simultánea de la tecla COMMODORE y la letra representada. Ej.: < M > =COMMODORE M.

– Las letras aparecidas entre barras verticales deberán ser introducidas como pulsación simultánea de la tecla SHIFT y la letra representada. Ej.:|K|= SHIFT K.

– Entre corchetes simples se representarán los símbolos que se obtienen por pulsación directa de la tecla, aunque lógicamente, este caso sólo se dará para indicar las sucesiones de más de una letra. Así por ejemplo, la introducción de 5 asteriscos se representaría por [5*].

– Para la repetición de símbolos obtenidos mediante las teclas COMMODORE o SHIFT, se seguirá una combinación de las tres normas anteriormente citadas. Así por ejemplo, la introducción de 10 símbolos COMMODORE H, se representaría por [ < 10 H > ].

– Para evitar confusiones, cuando se utilice el sistema de representación de sucesiones de carácter, y éste sea un espacio, se utilizará la abreviatura ESP. [15 ESP] = 15 espacios.

– Los caracteres de control, tales como desplazamientos del cursor, colores, estados de reversa y funciones, se simbolizarán por una abreviatura de tres letras (dos más un espacio en el caso de las funciones) encerradas entre llaves, tal como se señala en la tabla adjunta.

Para introducir cualquier listado por el sistema FUERA ERRORES, deberemos entrar previamente y ejecutar el listado BASIC que aparece en la página siguiente por lo cual es recomendable conservar una copia grabada del mismo, para sucesivas ocasiones. Una vez introducido este listado, ya sea por el teclado, o a través de cinta o disco, debemos ejecutarlo con **RUN.** Instantes más tarde aparecerá en la pantalla el mensaje **FUERA ERRORES!** y el cursor libre para la entrada de programas, con el tradicional **READY.** Por encima, lo cual indicará la activación del sistema de depuración de errores.

En virtud al **NEW** que finaliza línea 20 del programa ¡FUERA ERRORES!, éste habrá desaparecido de la memoria, y seremos libres para introducir cualquiera de los programas listados en la sección **TECLA A TECLA** de cualquier número de nuestra revista, o incluso aquellos que apareciendo en otras secciones se acojan a este sistema. Así pues, si el programa ¡FUERA ERRORES! ha desaparecido de la memoria, ¿qué hemos conseguido ejecutándolo? Bien, la respuesta se llama informáticamente **INTERRUPCION:** se trata de una técnica de programación en código máquina que permite que el ordenador efectúe prácticamente dos trabajos a un tiempo, o más correctamente, que ejecute determinada tarea de forma automática, sin necesidad de que le prestemos una atención constante, de forma similar al proceso de respiración en un humano.

Efectivamente, aunque el soporte BASIC ha desaparecido de la memoria, antes de «marcharse» ha dejado funcionando en modo interrupción la pequeña rutina en código máquina que se hallaba en sus **DATAS.** Para comprobarlo pulsa **RETURN;** observarás algo muy extraño: tu ordenador no se comporta normalmente, no sólo desciende una línea el cursor, sino que además hace aparecer un número en la esquina superior izquierda de la pantalla. A continuación veremos como emplearlo.

Cada vez que pulsemos **RETURN,** aparecerá un número en la mencionada zona de la pantalla, y éste corresponderá con la instrucción que hayamos introducido. Esto forma parte del sistema de FUERA ERRORES. Cuando introduzcamos cualquier línea de un listado de este tipo, deberemos fijarnos en el número que aparece al pulsar **RETURN** de fin de línea; si éste coincide con el que aparece en el listado al final de la línea, ésta habrá sido introducida correctamente, en caso contrario existe algún error de tecleado que debemos modificar. Para modificar una instrucción errónea, no tenemos ni tan siquiera que volver a teclearla si no queremos, bastará sencillamente con modificar el carácter o caracteres erróneos como siempre hacemos, hasta que coincida el número de verificación que se presentará al pulsar **RETURN.**

Así pues, el sistema FUERA ERRORES se compone de dos partes: una codificación especial de los listados que facilitan su introducción, evitando los errores al confundir los caracteres gráficos, de control, etc., y un sistema de verificación de líneas que nos advierte en el preciso instante de introducir una de estas, que está mal tecleada.

Ahora bien, el empleo de estos dos sistemas no quiere decir que nos encontremos ante un BASIC diferente al de COMMODORE 64. Este no ha cambiado, hace exactamente las mismas cosas de siempre; simplemente hemos cambiado la forma de hacer los listados. En cuanto al misterioso número que aparece en la esquina de la pantalla no es más que una simple suma de comprobación, lo que se conoce técnicamente como un CHECKSUM. La rutina en código máquina de interrupción suma los valores de los caracteres que entra-

**TABLA DE INTERPRETACION DE CODIGOS DE CONTROL**

| ABR | SIGNIFICADO | OBTENCION |
|---|---|---|
| HOM | HOME | CLR/HOME |
| CLR | CLEAR + HOME | SHIFT CLR/HOME |
| ABJ | CURSOR ABAJO | CRSR VERTICAL |
| ARB | CURSOR ARRIBA | SHIFT CRSR VERTICAL |
| DCH | CURSOR DERECHA | CRSR HORIZONTAL |
| IZQ | CURSOR IZQUIERDA | SHIFT CSRS HORIZONTAL |
| RON | REVERSE ON | CTRL 9 |
| ROF | REVERSE OFF | CTRL 0 |
| F1 | FUNCION 1 | F1 |
| F2 | FUNCION 2 | SHIFT F1 |
| F3 | FUNCION 3 | F3 |
| F4 | FUNCION 4 | SHIFT F3 |
| F5 | FUNCION 5 | F5 |
| F6 | FUNCION 6 | SHIFT F5 |
| F7 | FUNCION 7 | F7 |
| F8 | FUNCION 8 | SHIFT F7 |
| BLK | BLACK (NEGRO) | CTRL 1 |

| ABR | SIGNIFICADO | OBTENCION |
|---|---|---|
| WHT | WHITE (BLANCO) | CTRL 2 |
| RED | RED (ROJO) | CTRL 3 |
| CYN | CYAN (CIAN) | CTRL 4 |
| PUR | PURPPLE (PURPURA) | CTRL 5 |
| GRN | GREEN (VERDE) | CTRL 6 |
| BLU | BLUE (AZUL) | CTRL 7 |
| YEL | YELLOW (AMARILLO) | CTRL 8 |
| NRJ | NARANJA | COMMODORE 1 |
| MRN | MARRON | COMMODORE 2 |
| RCL | ROJO CLARO | COMMODORE 3 |
| GR1 | GRIS 1 | COMMODORE 4 |
| GR2 | GRIS 2 | COMMODORE 5 |
| VCL | VERDE CLARO | COMMODORE 6 |
| ACL | AZUL CLARO | COMMODORE 7 |
| GR3 | GRIS 3 | COMMODORE 8 |
| PI | π | SHIFT ↑ |
| ^ | ↑ | ↑ |
| \ | £ | £ |
| STP | STOP | RUN/STOP |
| INS | INSERT | INST/DEL |
| — | ← | ← |

mos en cada línea que introducimos, aplicándoles un módulo 256, es decir, volviendo a cero cada vez que su suma parcial supera el 255, de este modo se crea un número entre 0 y 255 dependiente directamente de los caracteres introducidos. Así pues, es prácticamente imposible que una sucesión de errores den por casualidad ese número, mientras que siempre que la línea esté correctamente introducida obtendremos el mismo código que nosotros hallamos por igual sistema y añadimos al final de cada línea al realizar el listado del programa.

Por tanto, el misterioso número no es tampoco una modificación del BASIC de COMMODORE, sino simplemente un pequeño truco para la comprobación de que las líneas han sido bien introducidas. Propiamente no nos evita cometer errores de tecleado, sino que simplemente nos advierte inmediatamente en qué línea los hemos introducido.

Para desactivar el sistema sólo deberemos pulsar **RUN/STOP+RESTORE**, y si por cualquier motivo nos interesara reactivarlo, podríamos ejecutar **SYS 822**, siempre y cuando se encuentre el código máquina en la memoria, lógicamente.

¡ADVERTENCIA! Puesto que el código máquina se encuentra ubicado en el buffer del casete, es imprescindible desactivarlo **(RUN/STOP+RESTORE)** antes de realizar cualquier operación con dicho periférico.

## Utilización del casete con fuera errores

Si queremos introducir parte de un programa, para continuar posteriormente el trabajo emprendido, sin perder por supuesto la enorme ventaja del FUERA ERRORES deberemos proceder de la siguiente forma:

1. Desactivar el sistema FUERA ERRORES mediante la pulsación de RUN/STOP + RESTORE.
2. Trasladar el código máquina desde su ubicación en el buffer del casete hasta otro punto desocupado en la memoria RAM (por ejemplo 49152). Para ello sólo hemos de saber que ocupa 114 bytes desde 822 (inclusive). Una sencilla rutina que efectúe el trabajo de reubicación puede ser: **FORI=0TO113:POKE49152+I,PEEK(822+I):NEXT.**
3. Realizar la operación correspondiente con el casete; ya sea grabar una copia de seguridad de lo introducido en la memoria hasta el momento, ya sea cargar el casete una parte ya grabada del programa.
4. Restablecer FUERA ERRORES a su punto original. Siguiendo el caso de la línea de ejemplo anterior: **FORI=0TO113:POKE822+I,PEEK(49152+I):NEXT.**
5. Reactivar el sistema FUERA ERRORES mediante SYS 822 (esta vez no aparecerá mensaje alguno, aunque al pulsar **RETURN** comprobaremos que aparecen las cifras de control en la esquina superior izquierda). Lógicamente, los pasos 4 y 5 no son necesarios si después de una grabación se va a apagar el ordenador, y sólo se llevarán a cabo después de las cargas, o si tras la grabación de una copia de seguridad del programa, deseamos continuar introduciéndolo acto seguido.

```
10 FORI=822TO935:READA:C=C+A:POKEI,A:NEXT
20 IFC<>15254THENPRINTCHR$(147)"ATENCION!, HAY UN ERROR EN LOS DATOS":END
30 PRINTCHR$(147)TAB(213)"FUERA ERRORES!":SYS822:NEW
100 DATA 169,3,141,37,3,169,69,141,36,3,169,0,133,254,96,32,87,241,133,251
110 DATA 134,252,132,253,8,201,13,240,13,24,101,254,133,254,165,251,166,252
120 DATA 164,253,40,96,169,13,32,210,255,165,214,141,176,3,206,176,3,169,0
130 DATA 133,216,169,18,32,210,255,169,19,32,210,255,169,45,32,210,255,166
140 DATA 254,224,100,176,5,169,48,32,210,255,224,10,176,5,169,48,32,210,255
150 DATA 169,0,133,254,32,205,189,169,45,32,210,255,173,176,3,133,214,76,88,3
```

# Tableta o ligth Pen, ¿quién da más?

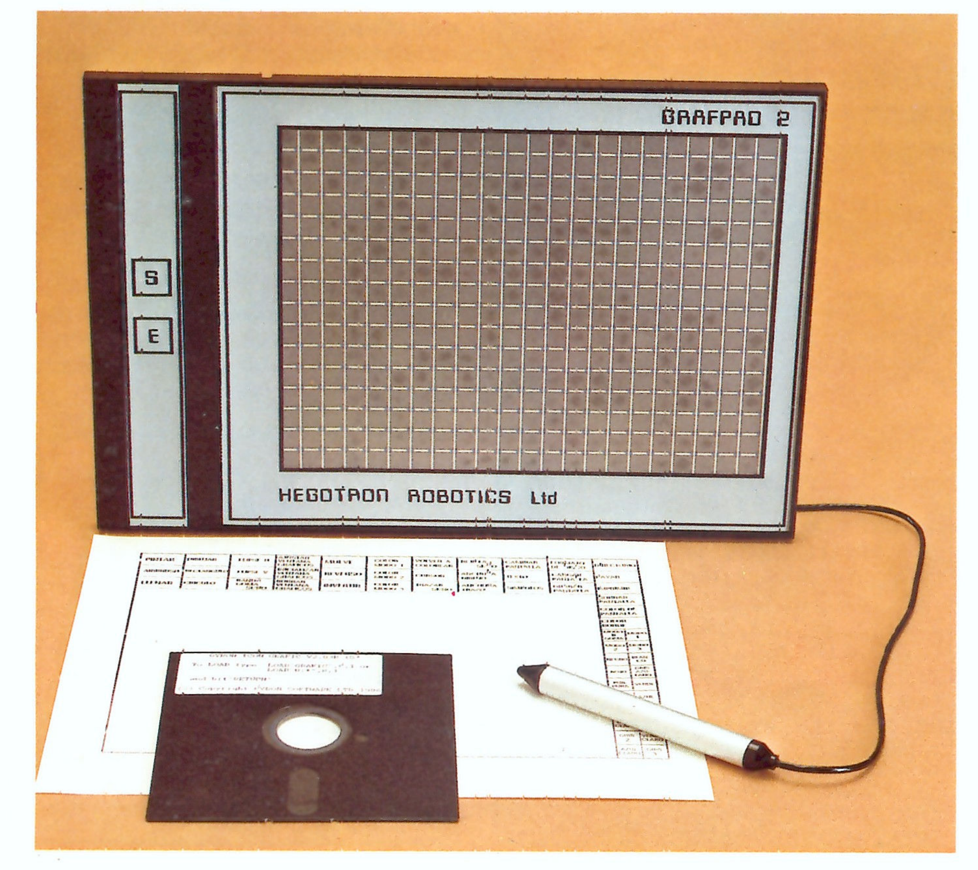
A pesar de ser dos periféricos conocidos por los lectores habituales de nuestra publicación, vamos hoy a redescubrirlos, en un enfrentamiento sano e instructivo, sobre todo para aquellos que estén interesados en la futura compra de uno de estos dispositivos gráficos.

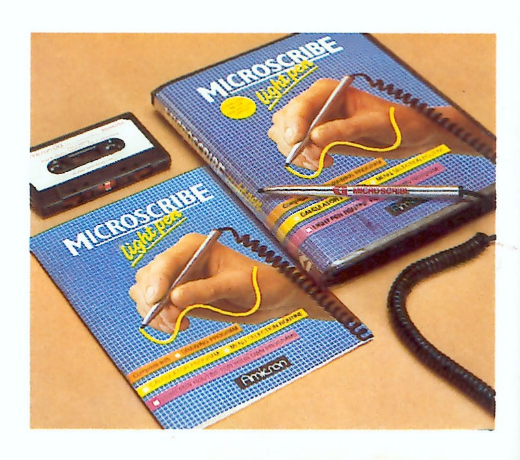
Los equipos que sometemos a exámen en nuestra sección de este mes son: el lápiz óptico «MICROSCRIBE light pen» de AMICRON, y la tableta gráfica «GRAFPAD II» de HEGOTRON. Como las definiciones nunca están de más, vamos a comenzar el artículo diciendo en qué consiste cada uno de estos periféricos.

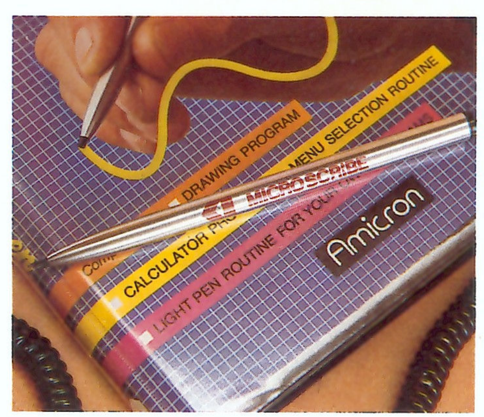
## El lápiz óptico

Es un dispositivo diseñado para crear gráficos en la pantalla. Está formado por una especie de bolígrafo, que sirve para dibujar directamente sobre la pantalla y el software necesario para su funcionamiento.

Dependiendo de la calidad del hardware empleado, así como de la del programa de control, tendremos un producto excelente o nada más que regular. El modo de funcionamiento se basa en el hecho del barrido de la pantalla, línea a línea, para la formación de la imagen; en el C64, hay una posición de memoria que nos dice qué línea se está dibujando en cada momento,



siendo capaz el lápiz óptico de mandar una señal, cada vez que coincide la línea dibujada con la posición de éste en la pantalla. Esta información, se utiliza en cada momento por el programa gráfico de ayuda, para efectuar la tarea de dibujar justo (o casi justo) en ese punto de la pantalla.

## La tableta gráfica

Al igual que el lápiz óptico, también éste es un dispositivo diseñado para la construcción de gráficos en la pantalla.

Está formado por una superficie plana sensible a la presión, al paso de una punta magnética, luminosa, etc., que envía al ordenador la posición relativa del utensilio utilizado (normalmente otra especie de bolígrafo), para que el programa correspondiente dibuje sobre la pantalla fielmente (o casi fielmente), nuestros movimientos sobre la tableta. Igualmente, aquí son muy importantes la calidad de hardware y software utilizados en su diseño.

## Gráficos en el C-64

Como todos sabemos, el C64 es un ordenador con una potencia gráfica respetable, pero como lamentablemente también conocemos, esa potencia se halla escondida en lo más recóndito de la memoria, siendo accesible tan solo a base de PEEKs y POKEs. Insistimos en este punto, porque todos los periféricos del tipo de los hoy comentados, vienen acompañados de un programa gráfico de ayuda, son importantísimos, ya que dependerá de ellos el que el periférico sea un instrumento útil o un juguete para hacer rayajos en la pantalla.

## Tableta gráfica Grafpad II

Comenzamos pues, con la tableta GRAFPAD II. En cuanto al aspecto externo del equipo cabe destacar que está montado sobre una carcasa de plástico duro, de dimensiones 24X36 cm, siendo su altura de tan solo 1.5 cm. La superficie útil de dibujo, está plastificada y ocupa un rectángulo cuadriculado de 18 x 25 cm.

Conectados a la parte inferior del mueble, salen dos cables: uno destinado al interface tableta-ordenador, y el otro para el «bolígrafo» de dibujo. El cable de interface, se conecta al port de usuario el C64 y para su utilización, es necesario cargar un programa de ayuda que se suministra en disco. Con ésto ya podemos empezar a dibujar.

El programa de ayuda facilitado para esta tableta, es de una calidad poco frecuente, y hace uso de la última moda informática sobre menús de opciones: los iconos.

Para seleccionar una de las muchas posibilidades que nos ofrecen (40 en total), deberemos dirigir el bolígrafo trazador a la parte inferior de la tableta, con lo que aparecerán en la misma zona de la pantalla dos filas con 20 iconos cada una, representativas de la opción; hay que hacer notar, que las opciones aparecen en forma de ventana, es decir, el trozo de dibujo que cubren al aparecer no es destruido cuando la ventana desaparece. Esta es otra de las últimas modas en informática que vienen en ayuda del sufrido usuario.

## Iconos: bienvenidos

En cada una de las 40 casillas que aparecen en la ventana de ayuda (de unos 15x15 pixels), aparecerá un dibujito alusivo (más o menos afortunado) a la tarea que realizará esa opción. Así, un rectángulo será la opción para dibujar cuadriláteros, una rejilla indicará si deseamos o no colocar un entramado en la pantalla, etc.

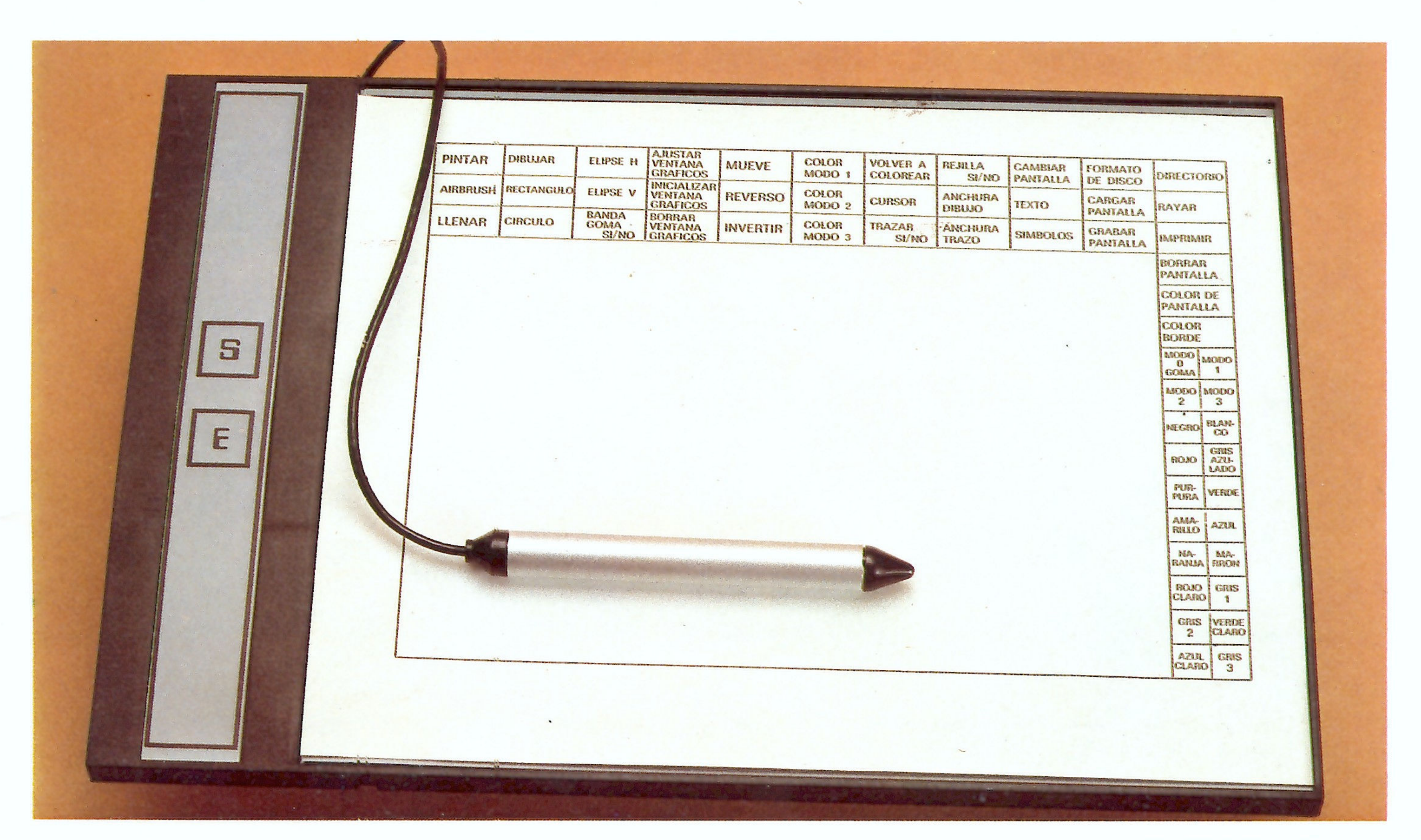
Entre las 40 opciones de que disponemos, cabe reseñar las más importantes:

— Trazar, sirve para dibujar de forma contínua.

— Spray, nos permite colocar diferentes entramados (24).

— Llenar, rellena zonas cerradas con tramas diferentes (24).

— Rectas, Triángulos, Rectángulos, Círcu-



los y Elipses, tienen una función obvia.

— Esquemas electrónicos, sirve para dibujar circuitos electrónicos, y cuenta con un editor de iconos predefinidos, parecido a un editor de sprites.

Además, dispone de opciones para el copiado de unas zonas de pantalla sobre otras e incluso de zoom, para ver en detalle ciertas áreas de pantalla.

Si deseamos escribir texto en cualquier punto, no será ningún problema con el icono correspondiente.

Podremos trabajar con dos pantallas a la vez, intercambiando una por otra (icono: hoja de papel doblada), o si lo deseamos, cambiar el dibujo de izquierda a derecha o de arriba hacia abajo.

La elección del ancho de brocha como el de los colores de papel y tinta es fácil, pudiendo seleccionar tanto el ancho (1-B) como el alto (1-8) de la brocha y el color que deseemos para cada plano.

Por último, disponemos de opciones que repiten lo último que hayamos dibujado, y por supuesto, de la posibilidad de grabar y cargar nuestros trabajos, así como toda la gama de comandos para disco. Y como no, también podremos ver nuestro trabajo sobre papel, si disponemos de una impresora MPS 801 o compatible.

Esta tableta dispone en su parte izquierda de dos botones etiquetados «S» y «E»; pues bien, con el «E» podemos escoger cualquier opción, pero el «S» es aparentemente inútil. No obstante, este último detalle, ha sido subsanado en versiones posteriores segun nos aseguran desde Ofite Informática.

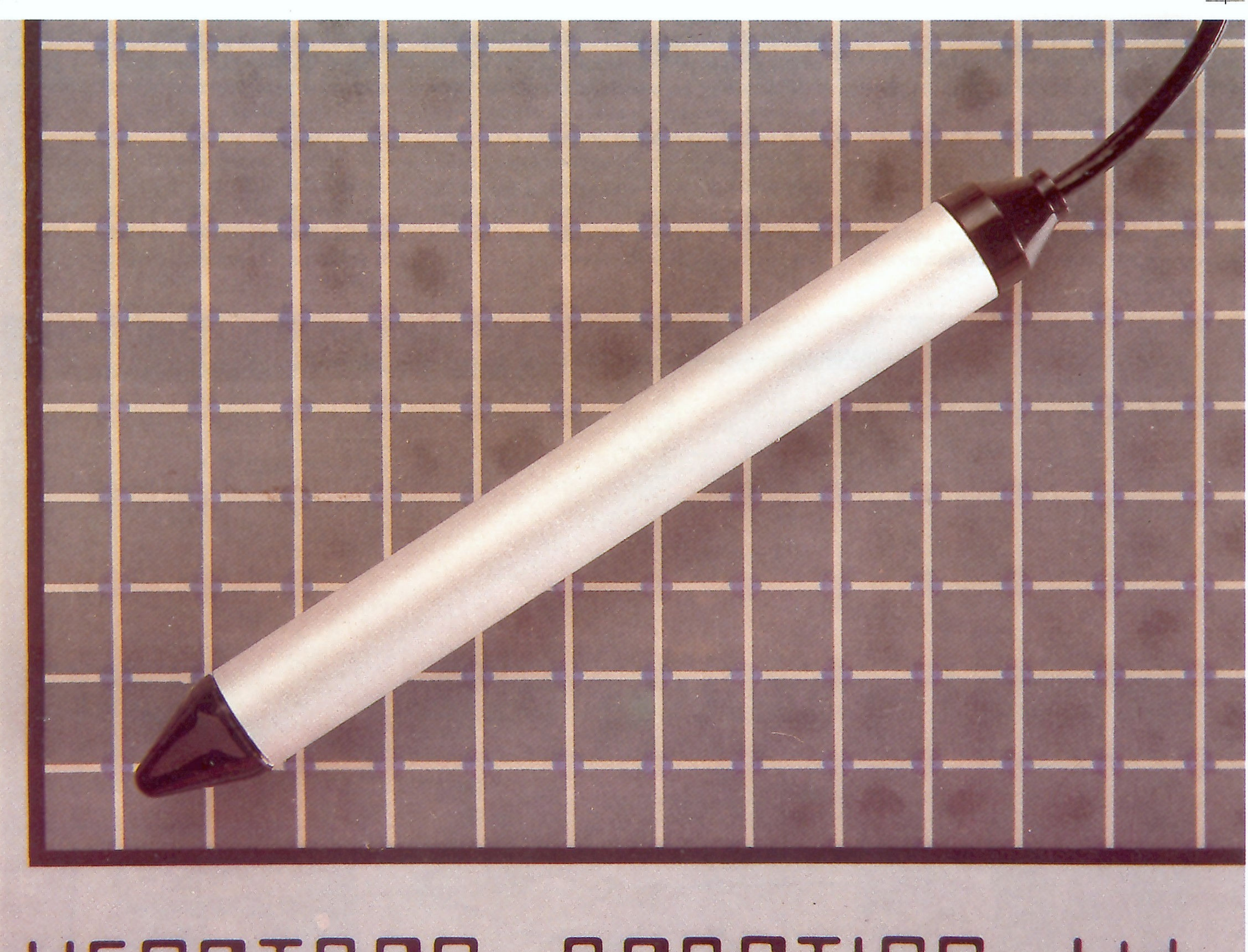
## Lápiz Optico Microscribe

Este periférico, mucho mas simple que la tableta gráfica, lo único que contiene es el «bolígrafo» trazador, con un cable en espiral que se conecta al port # 1, además de software correspondiente. Este último se suministra en una casete, pero cabe la posibilidad de grabar en disco todos los programas que alberga, al no encontrarse protegidos.

En la cara «A» del casete, tenemos dos programas principales: el Graffiti, que es el programa de dibujo propiamente dicho y el Pencalc, que es una calculadora de mesa (bueno, de pantalla), manejada a toque de lápiz. Aparte de estos programas, contiene el Penmenú, utilizado para elegir uno de los dos principales y el Pencode, alma del lápiz óptico, encargado del trabajo duro de posicionamiento y lectura. Todos los programas de la cara «A» están en turbo. En la cara «B» tenemos solamente el Graffiti con carga normal.

Graffiti, contiene seis opciones principales, que son:

DRAW: Sirve para dibujar y da acceso a cinco subopciones: Free, Poly, Brush, Lines y Fill. Con Free podemos dibujar libremente por la pantalla, donde vaya el lápiz irá el



trazo. Poly se utiliza para dibujar polígonos de 3 a 9 lados o círculos. Brush permite elegir la brocha a emplear, bien construyéndola nosotros mismos o eligiendo un trozo de pantalla ya dibujado. Lines nos permite elegir para el trazado de líneas, tres modalidades: unidas (final de una=principio de la siguiente), simples (damos el inicio y el fin de la línea) y rayos (damos un inicio y elegimos donde irán los finales). Fill sirve para colorear un área determinada con cualquier mezcla de los cuatro colores definidos (multicolor).

INKS: Permite definir los cuatro colores de trabajo, elegidos de cualquiera de los 16 disponibles. Además, tiene una opción etiquetada «I», que posibilita el cambio de los colores de lo dibujado, desde el color 1 al 2 o viceversa y desde el color 3 a fondo o viceversa.

ZOOM: Esta opción se utiliza para dibujar en detalle, ya que amplía la zona de pantalla elegida, cuatro veces. Podremos dibujar en cualquiera de los cuatro colores disponibles, manteniendo pulsado el lápiz contra la pantalla y soltando cuando el ciclo pase por el color deseado.

ERASER: Esta opción nos permite realizar dos tareas principales: borrar la pantalla por completo (Clear) o eliminar la última acción ejecutada (Restore). Esta última es particularmente útil cuando hacemos algo que no deseábamos (vamos, cuando metemos la patita), volviendo a las condiciones anteriores.

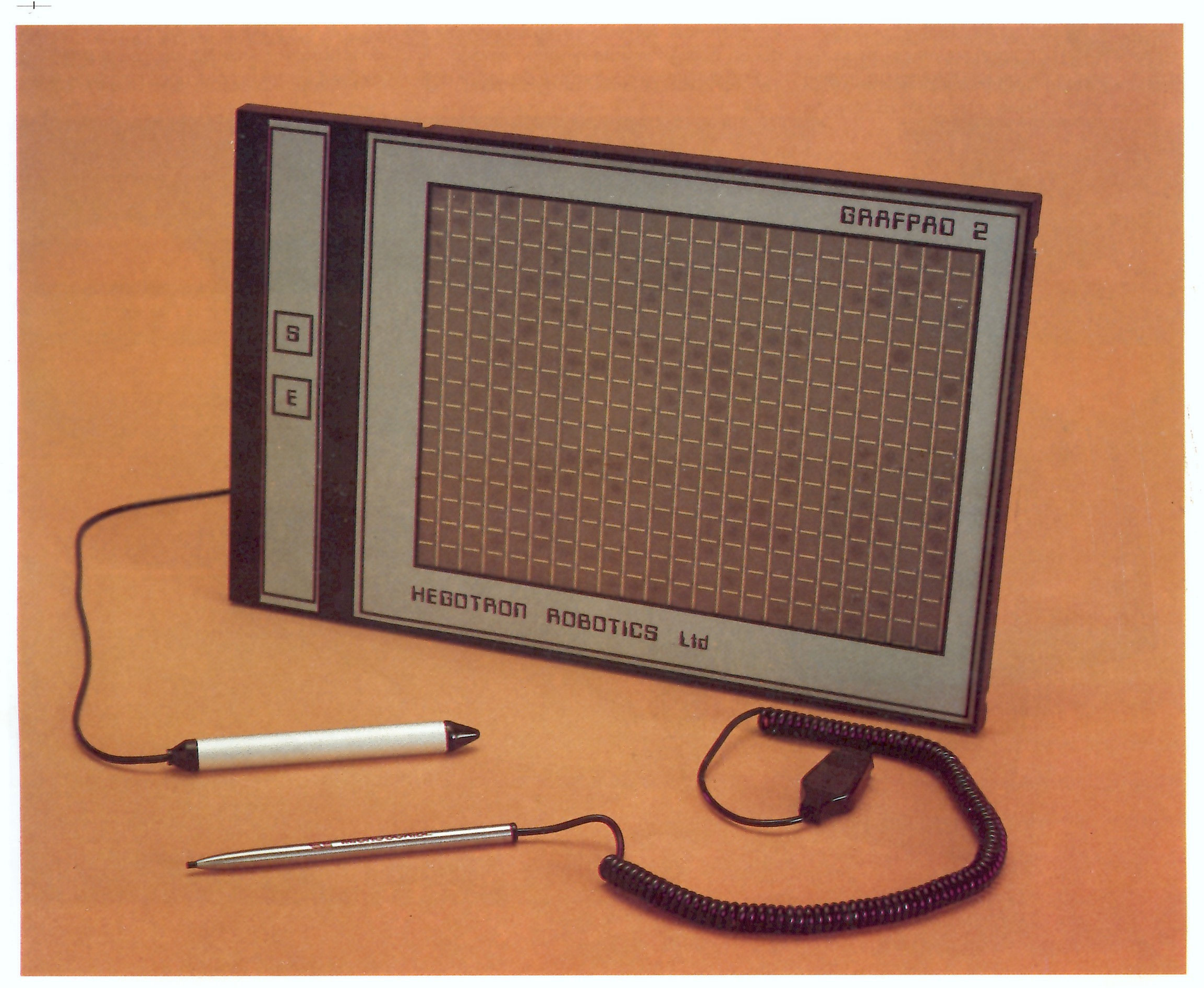
MEMORY: Sirve para Cargar, Grabar, e Ignorar, en Disco o Casete, con todas sus combinaciones; es decir, permite almacenar y/o recuperar pantallas en cualquier tipo de periférico. Si pulsamos la opción por error, Ignore la eliminará.

TYPE: Nos permite escribir texto en la pantalla, situándonos en cualquier punto de la misma. La introducción de texto se realiza por el teclado.

El programa Pencode dispone de todo tipo de ayudas, para su manejo en otras aplicaciones. En el manual se acompaña incluso un programa ejemplo para el manejo del lápiz óptico independientemente de los programas gráficos, disponiendo de ajustes (mediante POKEs) de la posición del sprite utilizado como puntero (el número 0), así como de ajustes de las coordenadas de salida. Todo un detalle hacia el usuario.

*Nombre:* Tableta Gráfica GRAFPAD II
*Distribuidor:* Ofites Informática, S.A.
Avda. Isabel II, 16
Tels.: (943) 45 55 44/45 55 33
28011 San Sebastián
*Precio:* 19.900 ptas.

*Nombre:* Lápiz Optico Microscribe
*Distribuidor:* Enfa Ibérica
Balandro, 39.
Tel.: (91) 742 18 92
28042 Madrid.
*Precio:* 10.000 ptas.



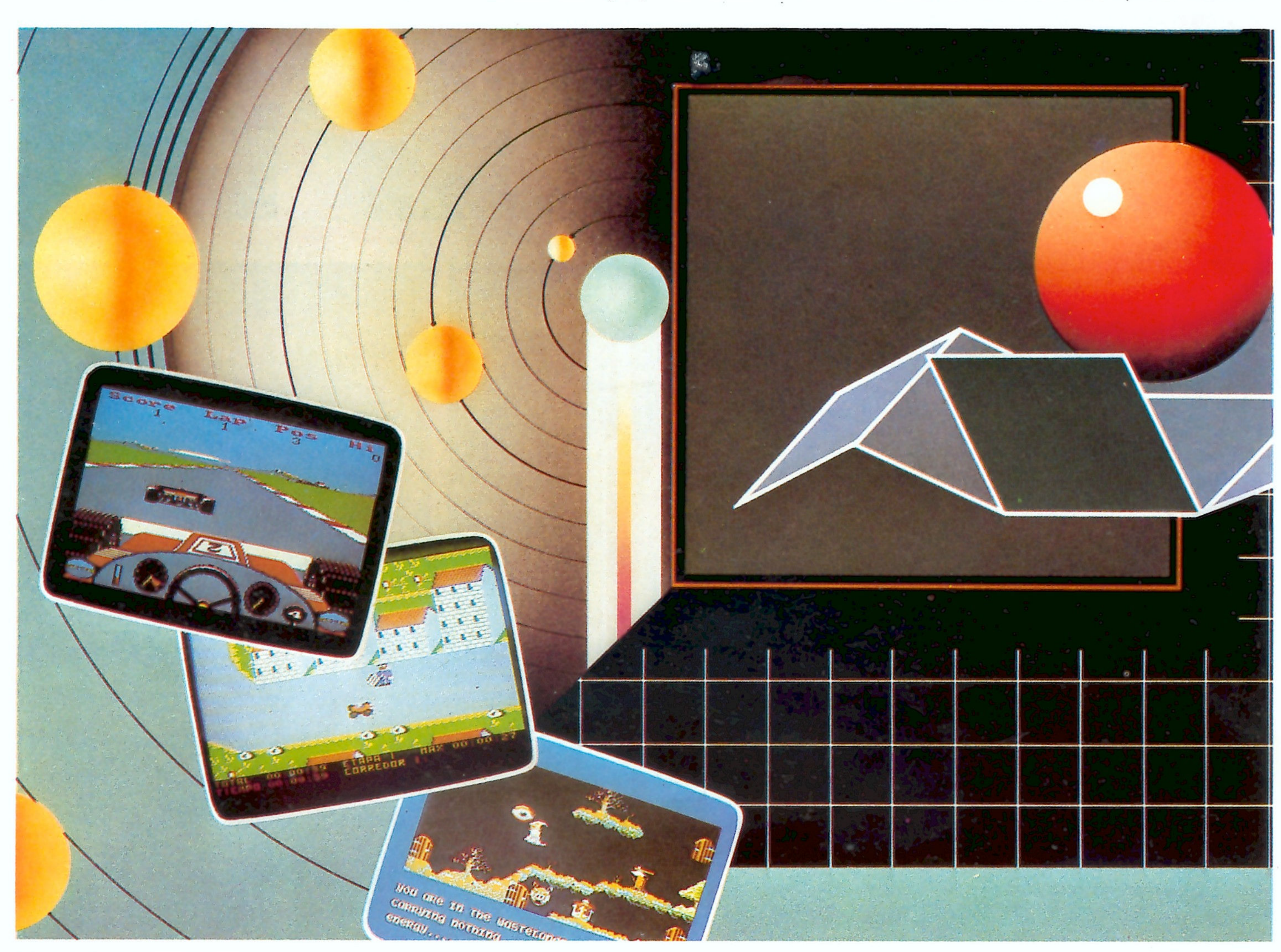
# Definiendo lo indefinido

Casi todos los ordenadores actuales, cuentan con un grupo de teclas independientes del teclado, que se denominan teclas de función. Normalmente, éstas llevan asociadas palabras clave que pueden ser fácilmente cambiadas a nuestro antojo. Este no es el caso de nuestro C64.

Cuando buscamos en el manual del C64 alguna información acerca de las teclas de función, la única conclusión que sacamos, es que son un grupo de teclas totalmente inútil, que sólo se pueden usar con una sentencia GET, exactamente igual que las demás. Pero en realidad, estas teclas tienen algo que las diferencia del resto: al pulsarlas, no producen absolutamente ninguna acción.

Esta distinción que parece desterrar definitivamente el uso de tan inútiles teclas, es precisamente la característica que hará que sean las más versátiles de todas, ya que con la debida programación, haremos que escriban lo que nos resulte más conveniente en cada momento, e incluso, que ejecuten pequeñas órdenes directamente.

Con el programa que hoy presentamos, vamos a definir 16 palabras que ejecutarán las teclas de función. Para conseguirlo, emplearemos las cuatro teclas principales (F1, F3, F5 y F7), y sus combinaciones con SHIFT, COMMODORE y CONTROL. Como podrás observar, en el listado del programa se definen las teclas con unas palabras esco-



gidas por el autor, pero puede darse el caso que tú normalmente uses otras mas a menudo; para redefinirlas a tu antojo, deberás proceder así:

Carga (o teclea, si es la primera vez) el programa del listado.

Modifica el contenido de las líneas DATA 230-380, ambas inclusive, respetando las siguientes reglas:

— No sobrepases en ningún caso, la longitud de cada palabra (9 caracteres), que viene determinada por las comillas de los **DATA.**

— No borres los espacios que sobren por la derecha, es decir, mantén siempre el **DATA** con 9 caracteres.

— Modifica sólo el primer dato de las **DATA,** deja intacto el segundo dato que se refiere a la tecla en cuestión (EJ: SH F1).

— Si lo deseas, puedes emplear caracteres de control (borrar pantalla, cambio de color, cursores, etc.), para ello, empieza a teclear abriendo las comillas del principio (encima de las anteriores) y teclea la nueva palabra. Ten en cuenta que cada carácter de control cuenta como uno.

— Para hacer que una acción se autoejecute al pulsar la tecla (como si pulsásemos RETURN al comando), añade al final el carácter flecha a la izquierda. Así por ejemplo: RUN (flecha izquierda), hará que al pulsar esa tecla, aparezca en pantalla la palabra RUN, y además, se ejecute. Este carácter también contará como uno.

—Utiliza las abreviaturas si deseas un comando largo, o incluso, combina dos o más teclas para que hagan una sola función. Por ejemplo, si quieres efectuar **FORI=1TO10:PRINTI:NEXTI+RETURN,** puedes definir una tecla (F1, pongamos por caso) con **FORI=1TO1** (9 caracteres), la tecla F3 con lo que sigue: **Ø=?I:N(SHIFT E)I(FLECHA IZQUIERDA)** que son otros nueve caracteres, y pulsar F1 y F3 para ejecutar la línea completa.

— Para conseguir el carácter comillas ("), sustitúyelo por el símbolo SHIFT+7 ('). El programa lo convertirá en comillas.

— Está prohibido terminantemente, emplear el carácter «libra» como parte del comando a definir, ya que lo utiliza el programa para saber dónde termina la frase de ayuda, que se verá posteriormente.

Una vez modificadas las líneas **DATA,** puedes grabar el programa con otro nombre diferente.

## *Nuestro programa*

En nuestra ayuda viene, ¿cómo no?, la socorrida interrupción IRQ (Interrup ReQuest). Desviaremos el conocido vector ($EA31 posiciones $0314 y $0315), para que rastree el teclado en busca de una de las cuatro teclas de función y en busca de las teclas especiales SHIFT, COMMODORE y CONTROL; al encontrar cualquiera de las combinaciones anteriores, mostrará en la pantalla la sentencia indicada en las líneas **DATA** del programa. Por lo tanto, tenemos disponibles 16 palabras a golpe de tecla; cómodo ¿no?

¿Cómo dices?, ¿que a ver quien es el guapo que se acuerda de lo que hay en cada una de las 16 combinaciones?, pues sí, tienes razón, para eso hemos implementado una rutina de ayuda, que te mostrará el contenido ACTUAL de cada tecla, y su obtención, bastará con que teclees SYS50000 (fácil ¿no?), para que aparezca una lista con todas las asignaciones. Los caracteres de control aparecerán tal y como aparecen entre comillas, y la tecla de RETURN con la consabida flecha izquierda. Si quieres, o eres de los que para sumar 2+2 enciendes el ordenador (es decir, vago como yo), puedes sacrificar una de las teclas (por ejemplo CTRL+F7) con la asignación SYS50000+RETURN (justo los nueve caracteres), para tener tu propio "HELP" a golpe de las teclas CTRL+F7.

Además, este programa es perfectamente compatible con el sistema FUERA ERRORES, permitiéndonos teclear más rápido los programas de nuestra revista (por lo menos cuando te acostumbres). Para usarlo conjuntamente con FUERA ERRORES, cárgalo y ejecútalo, teclea **NEW,** carga el FUERA ERRORES, ejecútalo y ya puedes teclear los programas cómodamente, con un mínimo esfuerzo.

## LISTADO

```
100 FORI=0TO15:READA$,B$:A$=A$+"  "+B$+"\":FORJ=1TOLEN
(A$)                                               -225-
110 K=ASC(MID$(A$,J,1)):IFK=ASC("_")THENK=13
                                                   -032-
120 IFK=ASC("'")THENK=34                           -123-
130 POKE49407+I*16+J,K:NEXTJ,I                     -020-
140 FORI=49152TO49407:READA:C=C+A:POKEI,A:NEXTI:POKE5
3281,0                                             -123-
150 FORI=50000TO50027:READA:C=C+A:POKEI,A:NEXTI:POKE5
3280,0                                             -097-
160 IFC<>41029THENPRINT"{CLR}{2 ABJ}ERROR EN LINEAS D
ATA":END                                           -028-
170 SYS49152:PRINT"{CLR}{PUR}TECLAS DE FUNCION ACTIVA
DAS"                                               -061-
180 PRINT"SYS5[4 0] PARA AYUDA"                    -086-
190 PRINT"SO= SOLO TECLA DE FUNCION"               -043-
200 PRINT"SH= SHIFT + TECLA DE FUNCION"            -168-
210 PRINT"CO= COMMODORE + TECLA DE FUNCION"        -199-
220 PRINT"CT= CONTROL + TECLA DE FUNCION{GRN}":SYS500
00                                                 -149-
```

```
230 DATA "{CLR}LIST [3 ESP]","SO F1"                        -106-
240 DATA "DATA[5 ESP]","SH F1"                              -144-
250 DATA "GOTO[5 ESP]","CO F1"                              -167-
260 DATA "INPUT[4 ESP]","CT F1"                             -228-
270 DATA "L:O:'$',8  ","SO F3"                              -082-
280 DATA "CHR$([4 ESP]","SH F3"                             -133-
290 DATA "GOSUB[4 ESP]","CO F3"                             -212-
300 DATA "PRINT[4 ESP]","CT F3"                             -222-
310 DATA "PEEK([4 ESP]","SO F5"                             -172-
320 DATA "TAB([5 ESP]","SH F5"                              -120-
330 DATA "AND[6 ESP]","CO F5"                               -100-
340 DATA "INPUT#[3 ESP]","CT F5"                            -234-
350 DATA "SAVE'[4 ESP]","SO F7"                             -187-
360 DATA "SPC([5 ESP]","SH F7"                              -141-
370 DATA "OR[7 ESP]","CO F7"                                -088-
380 DATA "PRINT#[3 ESP]","CT F7"                            -237-
390 DATA 120,162,13,160,192,142,20,3,140,21,3,88,96,1
65,197,201                                                  -083-
400 DATA 4,208,49,173,141,2,240,12,201,1,240,16,201,2
,240,20                                                     -151-
410 DATA 201,4,240,24,169,0,141,227,192,76,224,192,16
9,16,141,227                                                -188-
420 DATA 192,76,224,192,169,32,141,227,192,76,224,192
,169,48,141,227                                             -113-
430 DATA 192,76,224,192,201,5,208,49,173,141,2,240,12
,201,1,240                                                  -071-
440 DATA 16,201,2,240,20,201,4,240,24,169,64,141,227,
192,76,224                                                  -073-
450 DATA 192,169,80,141,227,192,76,224,192,169,96,141
,227,192,76,224                                             -122-
460 DATA 192,169,112,141,227,192,76,224,192,201,6,208
,49,173,141,2                                               -249-
470 DATA 240,12,201,1,240,16,201,2,240,20,201,4,240,2
4,169,128                                                   -254-
480 DATA 141,227,192,76,224,192,169,144,141,227,192,7
6,224,192,169,160                                           -214-
490 DATA 141,227,192,76,224,192,169,176,141,227,192,7
6,224,192,201,3                                             -107-
500 DATA 208,75,173,141,2,240,12,201,1,240,16,201,2,2
40,20,201                                                   -246-
510 DATA 4,240,24,169,192,141,227,192,76,224,192,169,
208,141,227,192                                             -101-
520 DATA 76,224,192,169,224,141,227,192,76,224,192,16
9,240,141,227,192                                           -207-
530 DATA 162,0,189,192,193,201,32,240,7,157,119,2,232
,76,226,192                                                 -147-
540 DATA 138,133,198,160,255,162,255,202,208,253,136,
208,248,76,49,234                                           -209-
550 DATA 169,1,133,212,162,0,189,0,193,201,13,208,2,1
69,95,201                                                   -033-
560 DATA 92,208,2,169,13,32,210,255,232,208,235,96
                                                            -226-

READY.                                                      -173-
```

# CURSO DE BASIC

## *PROGRAMACION DE GRAFICOS*

Hasta ahora hemos trabajado sobre la pantalla del ordenador en el modo carácter, es decir, almacenando sobre algunas de las 1000 posiciones posibles de ésta (40 filas por 25 columnas), algunos de los caracteres posibles, dentro del código del ordenador (de entre 256 posibles).

Este es, sin duda, el modo más simple, y a la vez el convencional, de dirigirnos a la pantalla del ordenador. Por otra parte, coincide con el único método con que podemos dirigirnos a ésta, desenvolviéndonos en el entorno de la sentencia BASIC **PRINT.**

Además, y basándonos en los conocimientos adquiridos anteriormente sobre la estructura de la pantalla, podemos dirigirnos a ella también por medio de la sentencia BASIC **POKE,** teniendo en cuenta la existencia de una «doble pantalla», en cuanto a su tratamiento con esta sentencia, por lo que, cada impresión de carácter, generará dos instrucciones **POKE** a la memoria de pantalla y de color, respectivamente.

Sin embargo, a base de estos dos sistemas, no conseguimos más que colocar determinado carácter o símbolo gráfico, en una «celda» concreta de la pantalla; no teniendo acceso a la que está «punto a punto». Este acceso puntual, denominado también «alta resolución», es el que permite dirigirse a cada uno de los puntos elementales de que se compone la pantalla.

Ya que cada código de carácter del ordenador se inscribe en el interior de una cuadrilla de 8 x 8 puntos, siendo posible almacenar hasta 1000 de éstos en la totalidad de la pantalla, podemos decir que, en un tratamiento punto a punto, tendremos acceso a 64000 posiciones elementales de



*Denominamos «alta resolución» al acceso puntual que permite dirigirse a cada uno de los puntos elementales de que se compone la pantalla.*

pantalla, lo que implica una «alta resolución», de gran utilidad en la representación de gráficos.

Pues bien, nuestro ordenador nos reserva una nueva sorpresa en éste sentido, ya que, además de permitirnos el acceso a la pantalla en alta resolución, nos facilita la definición de «sprites»: pequeños diseños gráficos de movimiento independiente dentro de la pantalla.

El artífice de esta alta capacidad de generación gráfica es el VIC II chip, (VIC 6567). Gracias a él es posible, por ejemplo, definir la mitad superior de la pantalla en alta resolución, la mitad inferior utilizarla para texto, y combinar los sprites con ambas mitades.

*En un tratamiento punto a punto, podremos acceder 64000 posiciones elementales de la pantalla.*

## DIFERENTES POSIBILIDADES

Dentro del modo de carácter estándar, es posible la utilización del juego de caracteres contenido en ROM, o la generación a partir de un juego definido por el usuario, empleando para ello parte de la memoria RAM. De forma similar, sucede con el modo carácter no estándar y con el modo de color de fondo extendido.

Dentro del modo «bit-map», disponemos de las opciones estándar y multicolor; de la misma manera que, en cuanto se refiere a la definición de sprites gráficos, pueden aplicarse estas mismas dos modalidades.

## POSICIONES DE LOS GRAFICOS

La pantalla del ordenador está compuesta por 1000 bytes de memoria consecutivos, los cuales se encuadran normalmente a partir de la posición 1024 y hasta la 2023. Cada uno de estos bytes dispone de 8 bits, lo cual permite almacenar cualquier entero en el rango desde 0 a 255 (para contener cualesquiera de los códigos de carácter disponibles en el ordenador).

Paralelamente a esta zona de memoria, existe otra gemela contenida desde la posición 55296 a la 56295. Cada una de estas posiciones está formada por 4 bits, permitiendo como contenido cualquier entero entre 0 y 15 (suficiente para albergar los códigos de color).

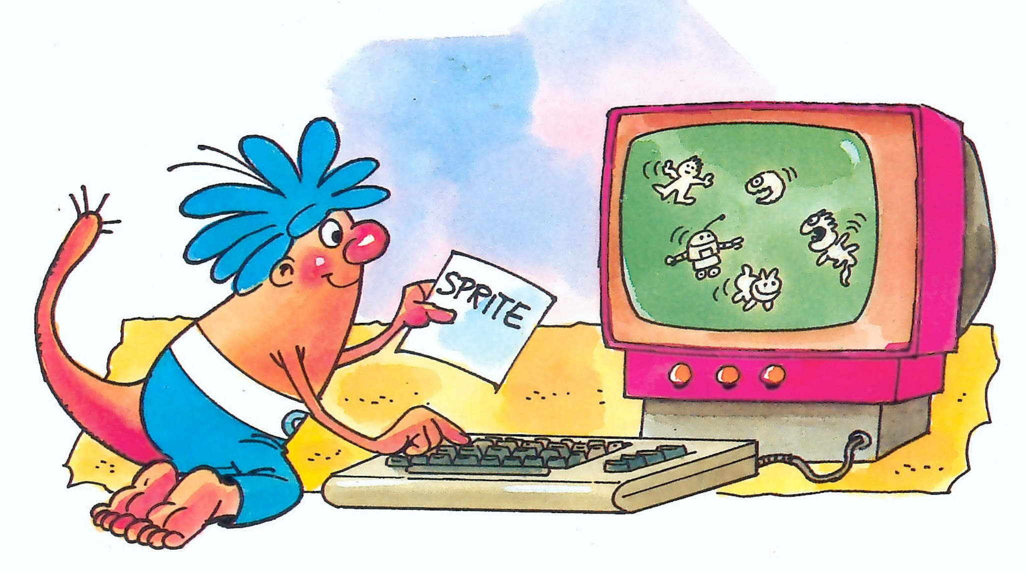
Dentro de esta estructura, los distintos modos gráficos se seleccionan mediante los 47 registros de control que posee el VIC-II. Para facilitar el diálogo con el VIC-II, podemos valernos de las sentencias BASIC **PEEK** y **POKE** a las posiciones de memoria que van desde la 53248 a la 53294.

## LOS BANCOS DE VIDEO

El microprocesador en el que está basada la CPU (Unidad central de proceso) de nuestro COMMODORE-64 es el 6510 y está dotado de un bus de direcciones de 16 bits y de datos de 8 bits. Por ello, sus operaciones elementales están limitadas al tratamiento de «palabras» de un byte, es decir, una única posición de memoria al tiempo, siendo capaz de direccionar 64 kbytes como máximo (0-65535 posiciones).

Sin embargo, no es éste el caso del 6567, el cual no es capaz de direccionar más de 16 Kbytes al tiempo. Por ello, debemos considerar la memoria de nuestro ordenador como dividida en 4 bancos de 16 kbytes cada uno.

Esta partición de la memoria no representa, sin embargo, problema alguno, ya que podemos seleccionar el banco de video con el que deseamos trabajar de manera sencilla; manipu-




*Los «sprites» son pequeños diseños gráficos de movimiento independiente dentro de la pantalla.*

lando mediante la sentencia **POKE** del BASIC la CIA 6526, más concretamente sus bits 0 y 1, de la posición de memoria 56576.

Para empezar, diremos que la siguiente sentencia inicializa el valor adecuado para esta posición:

**POKE 56578, PEEK (56578) OR 3**

Recordemos aquí que la utilización del operador **OR** se realiza a nivel binario. De esta forma, se efectúa un **OR** lógico entre el contenido del byte 56578 y la configuración binaria del 3 decimal, es decir, 00000011. Con ésto, conseguimos forzar al valor 1 los bits 0 y 1 de esta posición de memoria, sin alterar el contenido de los otros seis, 2 al 7.

Para seleccionar un banco de video concreto, podemos recurrir a la instrucción BASIC:

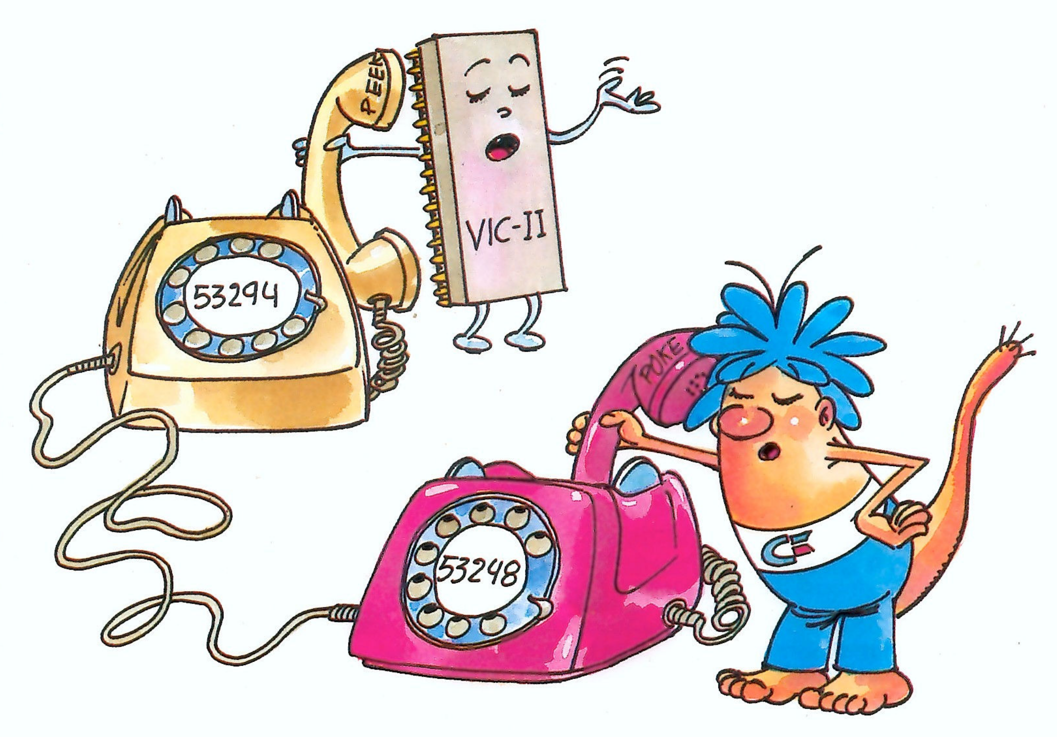
**POKE 56576, (PEEK 956576 AND 252) OR X%**

Representando la variable BASIC X% uno de los valores indicados en el siguiente cuadro:

| X% | BITS | BANCO | DESDE | HASTA |
|---|---|---|---|---|
| 0 | 00 | 3 | 49152 | 65535 |
| 1 | 01 | 2 | 32768 | 49151 |
| 2 | 10 | 1 | 16384 | 32767 |
| 3 | 11 | 0 | 0 | 16384 |

## *MEMORIA DE PANTALLA*

La posición inicial de la memoria de pantalla puede alterarse, también por medio de la sentencia BASIC **POKE**, en la posición de memoria 53272. Dentro de este byte, los 4 bits más significativos controlan la posición de comienzo de la memoria de pantalla. Por ello, podemos utilizar la siguiente instrucción de forma general:



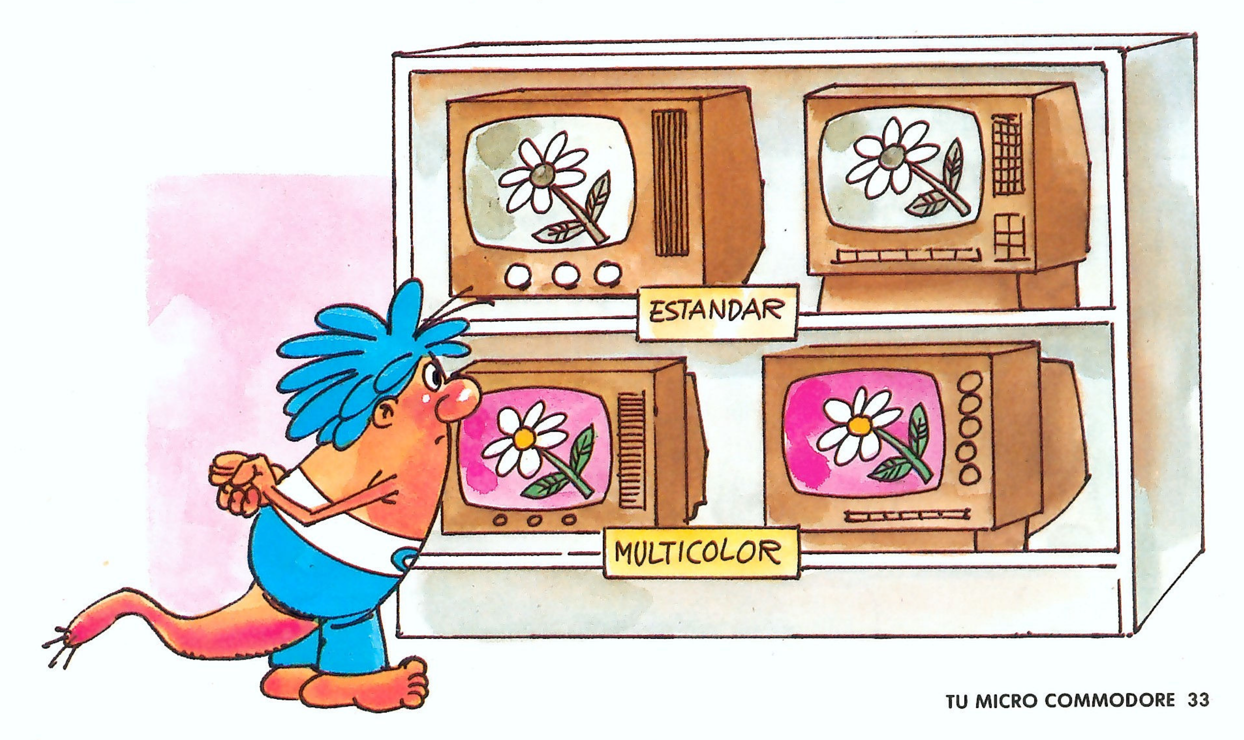

*Para facilitar el diálogo con el VIC-II, podemos valernos de las sentencias BASIC **PEEK** y **POKE** a las posiciones de memoria que van desde la 53248 a la 53294.*

**POKE 53272, (PEEK 53272) AND 15) OR X%**

Representando la variable BASIC uno de los valores indicados en el siguiente cuadro:

| X% | BITS | K. | DESDE | HASTA |
|---|---|---|---|---|
| 0 | 0000 | 1 | 0 | 1023 |
| 16 | 0001 | 2 | 1024 | 2047 |
| 32 | 0010 | 3 | 2048 | 3071 |
| 48 | 0011 | 4 | 3072 | 4095 |
| 64 | 0100 | 5 | 4096 | 5119 |
| 80 | 0101 | 6 | 5120 | 6143 |
| 96 | 0110 | 7 | 6144 | 7167 |
| 112 | 0111 | 8 | 7168 | 8191 |
| 128 | 1000 | 9 | 8192 | 9215 |
| 144 | 1001 | 10 | 9216 | 10239 |
| 160 | 1010 | 11 | 10240 | 11263 |
| 176 | 1011 | 12 | 11264 | 12287 |
| 192 | 1100 | 13 | 12288 | 13311 |

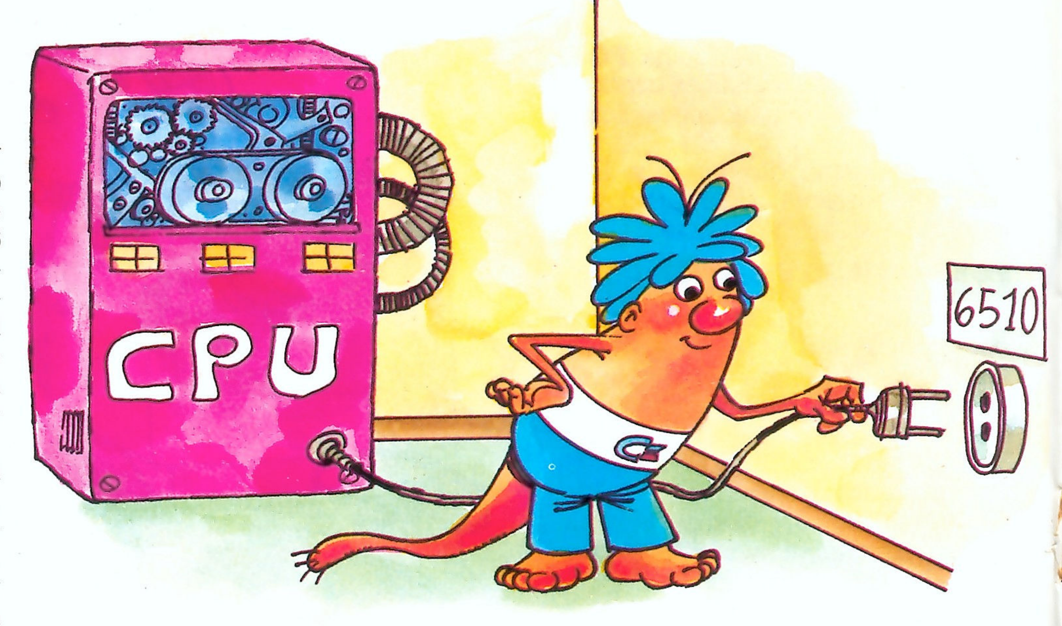
*Dentro del modo «bit-map» disponemos de las opciones estándar y multicolor.*



| | | | | |
|---|---|---|---|---|
| 208 | 1101 | 14 | 13312 | 14335 |
| 224 | 1110 | 15 | 14336 | 15359 |
| 240 | 1111 | 16 | 15360 | 16535 |

En cualquier caso, hemos de hacer notar que este nuevo posicionamiento de la memoria de pantalla, se inscribe dentro de uno de los rangos de 16 Kbytes. (bancos de video), comentados con anterioridad. Por ello, deberemos sumar este desplazamiento al indicado por el número de banco de video, de cara a obtener la dirección real de comienzo de la memoria de pantalla.

Otro punto a tener en cuenta, es el hecho de que la memoria de color, debido a sus especiales características de configuración (bytes de 4 bits), no es susceptible de reubicación. Por ello, debemos tener cuidado a la hora de escribir la nueva relación existente entre las posiciones de la memoria de pantalla reubicada y las invariables de la memoria de color.




*El microprocesador en el que está basada la CPU (Unidad Central de Proceso) de nuestro Commodore-64 es la 6510.*

## JUEGOS DE CARACTERES

Cada carácter a representar en la pantalla del ordenador está formado por una matríz de 8 filas por 8 columnas. Esto hace un total de 64 puntos posibles, de los cuales algunos deben permanecer «encendidos» (tomar el color del primer plano), o «apagados» (tomar el color del fondo de la pastilla), dando lugar al carácter en sí, tal como lo vemos.

Esta información no puede ser almacenada en menos de 8 bytes (de 8 bits cada uno), uno por cada fila de que se compone el carácter. Siguiendo esta norma, nos es evidente que, para contener la definición del juego completo de 256 caracteres de que dispone el COMMODORE-64, son precisos 2048 bytes (2 kbytes).

Inicialmente, el juego de caracteres del ordenador se encuentra contenido en la memoria ROM, ocupando 2 kbytes como ya hemos visto. No obstante, y ésta es una cualidad importante, podemos definir nuevos juegos de caracteres que se adapten a nuestras necesidades más concretas, siempre a costa de perder algo de la memoria RAM disponible.

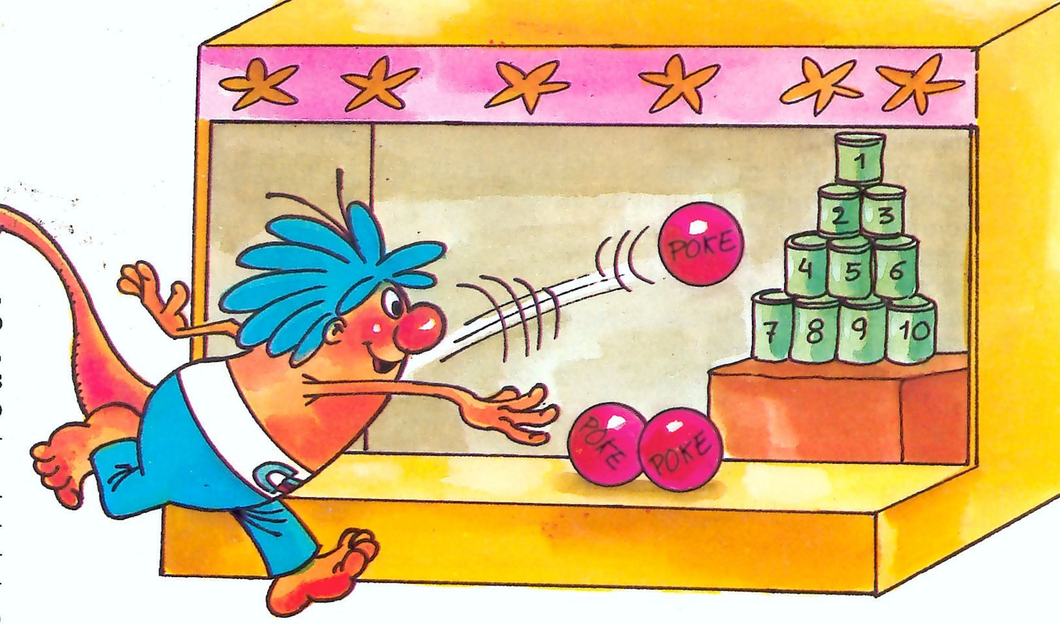
Sin embargo, debido a la limitación impuesta por el VIC-II, sólo sería posible trabajar con 8 juegos diferentes a un mismo tiempo, pues ocuparían la totalidad de los 16 kbytes que es capaz de direccionar.

El VIC-II necesita también conocer el lugar donde encontrar la redefini-

*Podremos seleccionar el banco de video manipulando mediante la sentencia **POKE** del BASIC la CIA 6526.*



*La posición inicial de la memoria de pantalla, puede alterarse también por medio de la sentencia **POKE.***




ción del juego de caracteres en RAM, en el caso de que hayamos hecho uso de ella; aunque sea parcialmente, puesto que en algunas ocasiones nos bastará con redefinir parte del juego de caracteres, únicamente los números y mayúsculas, etc...

En cualquier caso, es también posible que deseemos definir tan solo algunos caracteres especiales, manteniendo el juego básico del COMMODORE como estaba. En esta situación, se nos ofrece la posibilidad de copiar el juego íntegro contenido en ROM a una zona hábil de la memoria RAM, mediante un bucle del tipo **FOR NEXT,** para después redefinir por medio de **POKE** el o los caracteres concretos, sobre alguno de poco uso del código estándar.

El desplazamiento de comienzo del generador de caracteres se indica por medio de los bits 1, 2 y 3 (ignorando el 0) de la posición de memoria 53272, puesto que sólo son posibles ocho valores para cada banco de video de 16 kbytes seleccionados.

Recordemos que los cuatro bits más significativos de esta misma posición de memoria son los que informan al VIC-II del comienzo de la memoria de pantalla. Por ello, como en ocasiones anteriores, debemos recurrir a un algoritmo que nos permita influir solamente sobre el contenido de algunos bits, sin alterar el de los demás:

**POKE 53272, (PEEK (53272) AND 240) OR X%**

Donde X% debe tomar alguno de los valores sugeridos en el cuadro adjunto:

| X% | BITS | K | DESDE | HASTA |
|---|---|---|---|---|
| 0 | XXXX000X | 1 | 0 | 2047 |
| 2 | XXXX001X | 2 | 2048 | 4095 |
| 4 | XXXX010X | 3 | 4096 | 6143 |
| 6 | XXXX011X | 4 | 6144 | 8191 |
| 8 | XXXX100X | 5 | 8192 | 10239 |
| 10 | XXXX101X | 6 | 10240 | 12287 |
| 12 | XXXX110X | 7 | 12288 | 14335 |
| 14 | XXXX111X | 8 | 14336 | 16383 |

Debemos hacer notar que, este nuevo posicionamiento del generador de caracteres, implica sumar el posible desplazamiento al indicado por el número de banco de video, de cara a obtener la dirección de comienzo.

El generador de caracteres contenido en la ROM se encuentra en las posiciones 53248 a 57343. Sin embargo, dada la limitación del ámbito de la aplicación del VIC-II a 16 kbytes, existe una «imagen» en RAM de este generador de caracteres en ROM. Imagen, que no impide, por otro lado, que estas posiciones puedan ser ocupadas por los programas o datos del usuario.

De esta forma, el juego de caracteres ROM aparece emulado en las posiciones de RAM 36864 a 40959, dentro del banco 2, y en 4096 a 8191 dentro del banco 0. Este «reflejo» afecta, sin embargo únicamente a los bancos pares (0 y 2). Por ello, en el caso de producirse alguna interferencia con los caracteres gráficos generados por el usuario, podemos optar por situar el




*La memoria de color, debido a sus especiales características de configuración, no es susceptible de reubicación.*

juego de caracteres redefinido en los bancos impares, los cuales no tienen este problema.

En principio, esta especie de «compartición» de la memoria RAM puede parecernos chocante, sobre todo porque sabemos que una de las características de la memoria del ordenador es que, en cada una de sus posiciones, sólo puede albergarse un único contenido al mismo tiempo. Aparentemente, las posiciones ocupadas por el generador de caracteres ROM, coinciden con los registros de control del VIC-II. Esto es posible porque no lo hacen exactamente al mismo tiempo.

La explicación de este hecho reside en que, cuando el VIC-II necesita acceder al generador de caracteres, la memoria ROM que los contiene genera una «imagen» gemela en el banco de 16 kbytes. al que tiene acceso el VIC-II en ese momento, permaneciendo esta área ocupada por los registros de control de entrada/salida y quedando el generador de caracteres ROM únicamente disponible para el VIC-II.

En cualquier caso, ya hemos dicho que lo normal a la hora de necesitar redefinir el juego de caracteres, es que la redefinición sea parcial. Para ello, necesitamos copiar previamente los caracteres estandar contenidos en ROM en otras posiciones de memoria.

Este proceso de copia resulta algo complicado. Para empezar, debemos desactivar los registros de entrada/salida y activar la ROM del generador de caracteres, antes de realizar la copia, total o parcial, por medio de un bucle **FOR-NEXT.** Terminado este proceso, es necesario volver a activar los registros de entrada/salida.

Durante el tiempo de copia no se permite la interrupción del programa, por estar desactivados los registros de entrada/salida, no debiéndose efectuar tampoco, en este intervalo, la lectura del teclado, el cual ha de quedar inhabilitado. Para hacer esto, podemos recurrir a la instrucción BASIC:

**POKE 56334, PEEK (56334) AND 254**

Para volver a activar el teclado, una vez concluido el proceso de copia, puede escribirse:

**POKE 56334, PEEK (56334) OR 1**

Por otra parte, para desactivar los registros de entrada/salida y activar la imagen ROM del generador de caracteres, debemos recurrir a:

**POKE 1, PEEK (1) AND 251**

Por último, para retornar a la posición normal los registros de entrada/salida:

**POKE 1, PEEK (1) OR 4**



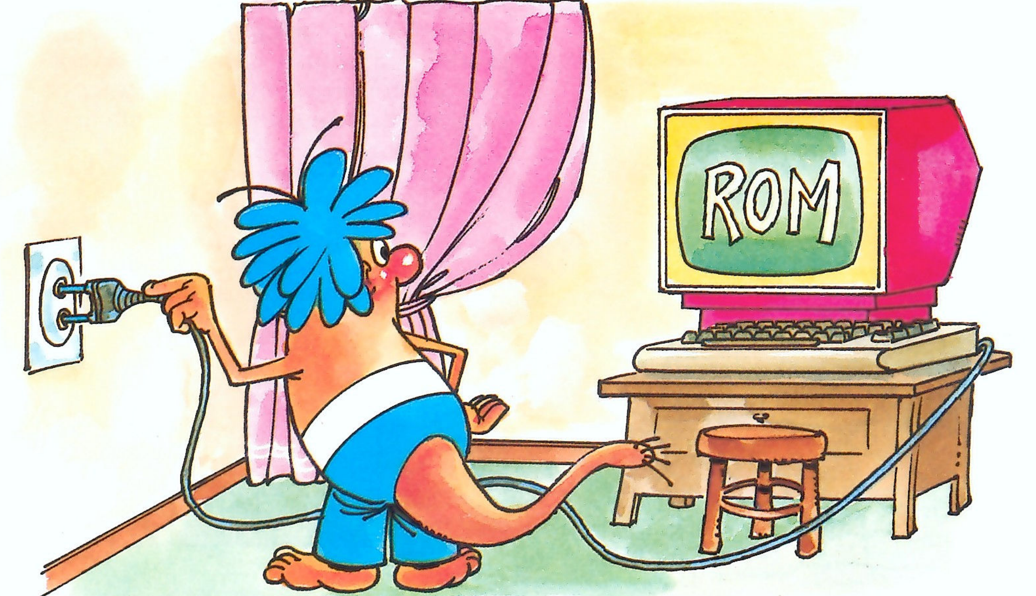
*Podemos definir nuevos juegos de caracteres a costa de perder algo de la memoria RAM disponible.*

## DEFINICION DE CARACTERES

Al conectar el ordenador, éste se encuentra en el modo carácter estándar, el cual nos permite acceder al generador de caracteres ROM.

Sin embargo, en caso de que deseemos alterar este estado de cosas, bien por desear un «cambio» en el tipo de letra, para una determinada aplicación, o bien para definir algunos o todos los caracteres, para incluirlos en un programa de juego, es necesario recurrir al procedimiento de traslado antes descrito.

Para definir caracteres es preciso, pues, el haber trasladado previamen-



*Al conectar el ordenador, éste se encuentra en el modo carácter estándar, el cual nos permite acceder al generador de caracteres ROM.*




*El primer carácter configurado en el generador de caracteres ROM es la arroba (@).*

te el juego contenido en ROM a alguna zona de la RAM. Una vez allí, debemos calcular la posición concreta en que se encuentra el código o los códigos de carácter, cuya configuración de puntos deseamos alterar.

Conseguir ésto no es difícil; nos basta con conocer algunas cosas sobre la configuración del área del generador de caracteres. De momento, sabemos que consta de 256 bloques de 8 bytes cada uno, ocupando un total de 2 kbytes., en su conjunto.

Cada uno de estos bloques de 8 bytes, se corresponde con la configuración binaria de cada una de las ocho filas que componen la definición de un carácter.

De este modo, los bits a 1 indican los puntos que deben permanecer encendidos, es decir, tomar el color determinado para el primer plano de la impresión; y los que toman el valor 0, representan a los que deben permanecer apagados, es decir, hacerse invisibles por tomar el mismo color que el definido para el fondo de la pantalla, o segundo plano de impresión.

La forma en que el Sistema localiza la forma de representar un carácter podemos sintetizarla en la fórmula:

**D%=B%*16384+J%*2048+C%*8**

Donde D% representa la dirección de comienzo de almacenamiento de la configuración del carácter, B% el banco de video, J% la posición relativa dentro del banco del juego de caracteres, y C% el código de carácter contenido en la memoria de pantalla.

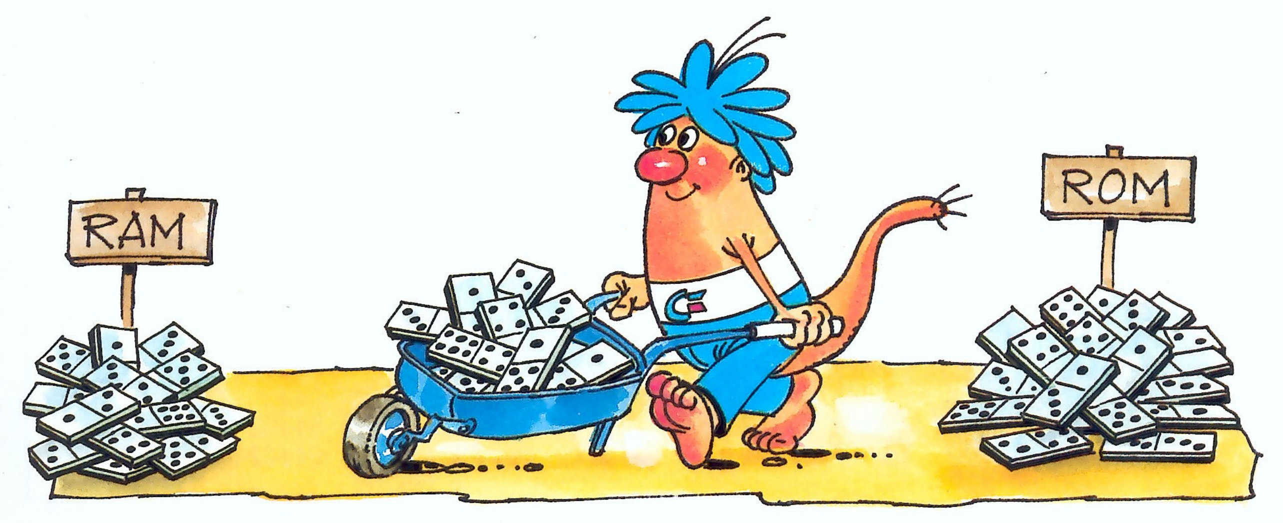
El primer carácter configurado en el generador de caracteres ROM es la «O» (código de pantalla 0), a continuación, se encuentra la representación del alfabeto en mayúsculas (de la «A» a la «Z»).

Puesto que el juego de caracteres comienza en la posición 53248, es evidente que saltando los 8 primeros bytes, tendremos acceso a la configuración de la letra «A» (posiciones 53256 a 53263). La representación de este carácter, resulta como sigue:

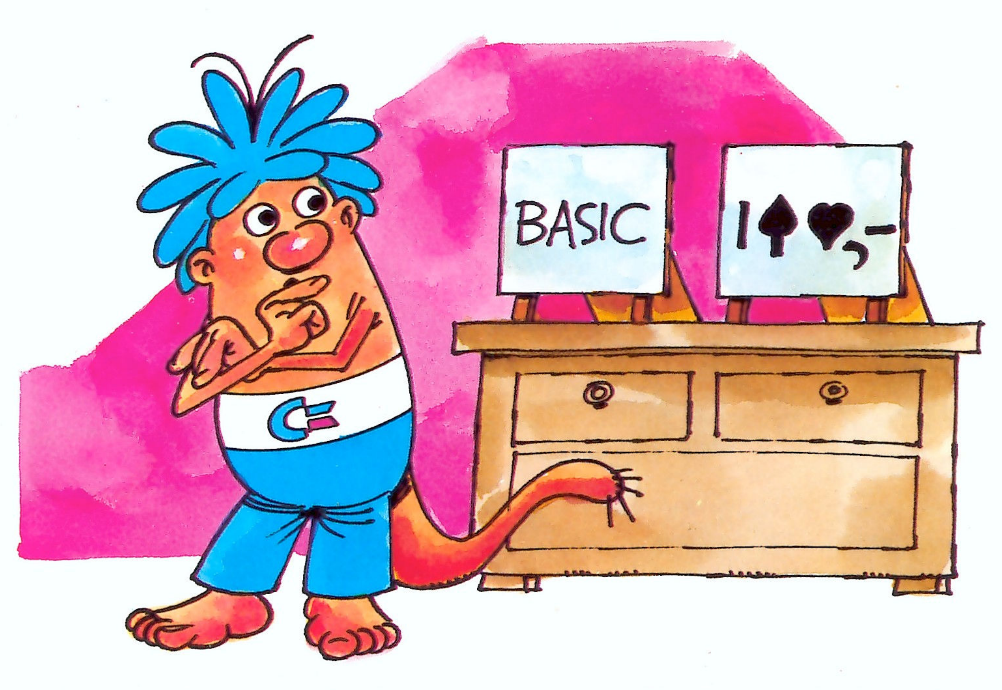
| BINARIO | DEC | CARACTER |
|---|---|---|
| 00011000 | 24 | ...**... |
| 00111100 | 60 | ..****.. |
| 01100110 | 102 | .** ..**. |
| 01111110 | 126 | .******. |
| 01100110 | 102 | .**..**. |
| 01100110 | 102 | .**..**. |
| 01100110 | 102 | .**..**. |
| 00000000 | 0 | ........ |

En el dibujo podemos ver claramente como se corresponde cada byte con cada linea de composición del carácter. De hecho, aunque la representación decimal de cada byte no nos da una idea concreta del carácter, al traducir este valor a su forma binaria, nos ha bastado con sustituir los ceros por puntos y los unos por asteriscos, para acceder a una clara representación, tal y como lo vemos en pantalla.

*Para definir caracteres es preciso trasladar el juego contenido en la ROM a alguna zona de la RAM.*



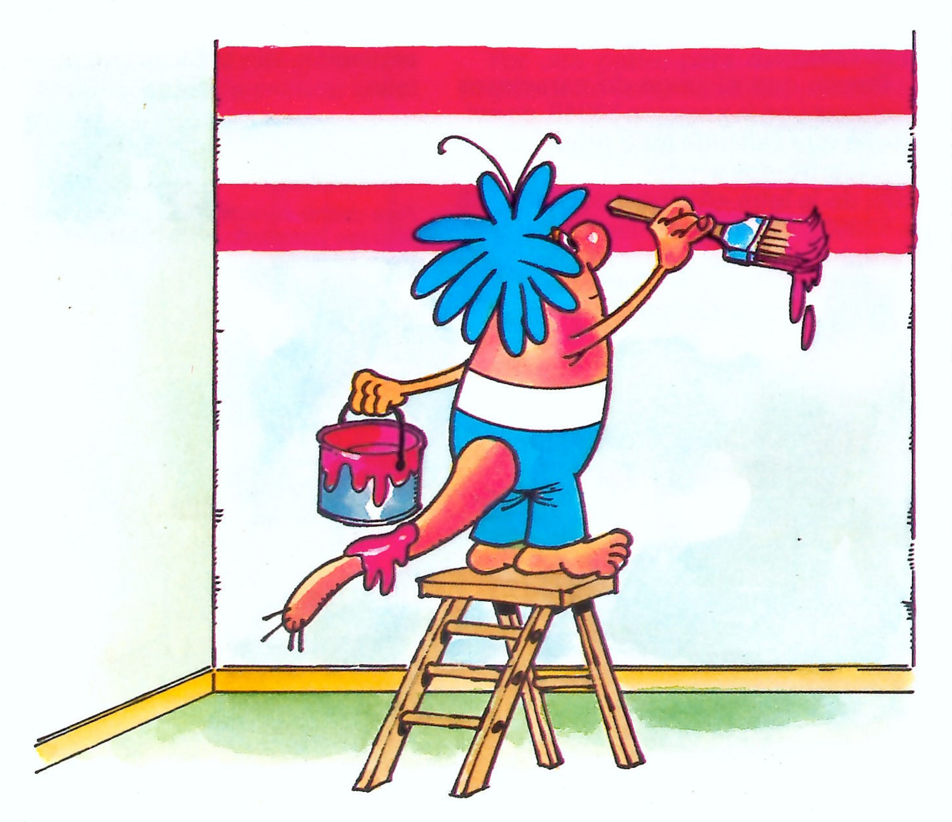
*El tamaño del juego de caracteres ROM es kbytes: 2 para el modo mayúsculas-gráficos y otras 2 para mayúsculas-minúsculas.*




Si nos fijamos un poco, y esta norma es aplicable a todo el juego de caracteres ROM, nos daremos cuenta de que cada carácter deja la fila inferior en blanco (a cero), para facilitar la separación de cada línea de impresión en pantalla de la inmediatamente inferior. Además, podemos ver también como las líneas verticales de los caracteres son de doble ancho, para facilitar la lectura de los mismos.

Otra consideración que debemos hacer, en cuanto al juego de caracteres ROM se refiere, es la de que tu tamaño es doble, es decir, 4 Kbytes, puesto que aparece desdoblado de forma que en los primeros 2 kbytes se encuentra el juego correspondiente a las mayúsculas y los símbolos gráficos, mientras que en los segundos 2 kbytes aparecen las representaciones de las letras minúsculas y la repetición de las mayúsculas.

*En el juego de caracteres ROM, cada carácter deja la fila superior en blanco, para facilitar la separación de cada línea de impresión en la pantalla.*



Esto es así, aunque redunde en una mayor ocupación de memoria ROM, para facilitar que con la pulsación de las teclas **COMMODORE** y **SHIFT** de forma simultánea, pueda pasarse directamente la pantalla de un modo de representación (mayúsculas/gráficos) a otro (minúsculas/mayúsculas).

De cara a generar nuestros propios caracteres gráficos, lo primero que debemos hacer es recurrir a una cuadrícula de 8 por 8 posiciones para, dentro de ella, marcar los puntos que deseamos aparezcan visibles (del color especificado para el primer plano de impresión) e invisibles (color del segundo plano), en formas diferentes.

De este modo, deberemos considerar como ceros binarios los puntos invisibles y unos los visibles. Sin embargo, y como en la mayoría de los casos nos será necesario recurrir a la representación decimal de los 8 bytes, de cara a utilizar la sentencia BASIC **POKE.** Nos será útil colocar, como cabecera de las ocho columnas de la malla, de izquierda a derecha, los números 128, 64, 32, 16, 8, 4, 2 y 1.

Estos números nos dan el valor «posicional» de cada uno de los bits «encendidos» (puntos visibles). De esta forma, el número decimal que se corresponde a cada serie de 8 bits binarios, podemos calcularlo sumando, siempre dentro de cada fila, el número indicado en la cabecera de cada columna cuando existe un bit 1 (punto encendido), únicamente.

Para mayor claridad, veamos como traduciríamos a valores decimales la representación binaria de la letra «A»:

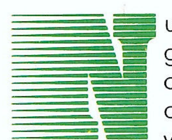
```
1
2631
84268421

...**...:              16+8=  24
..****..:         32+16+8+4=  60
.**..**.:         64+32+4+2=102
.*****.:   64+32+16+8+4+2=126
.**..**.:         64+32+4+2=102
.**..**.:         64+32+4+2=102
.**..**.:         64+32+4+2=102
........:=                      0
```




# Marble Madness

El ser humano siempre se ha sentido atraído por los juegos competitivos, en los que hay que apostar, o en los que uno de los jugadores, ayudado por su pericia y dominio de la técnica, doblega a todos sus competidores; del mismo modo, el niño desde que tiene uso de razón, lucha por ser mejor que el resto de sus compañeros en alguna faceta: es un instinto natural. Pues ahora tendrás ocasión de desarrollarlo en Marble Madness.



Nunca te has puesto en el lugar de una canica?, es un oficio muy sacrificado, ya que puedes ser un día la favorita de tu dueño gracias a un buen comportamiento en una carrera y al siguiente ser desahuciado por una mala racha y pasar a otras manos en el vasto mundo sumergido del mercado negro de las canicas. Allí te pueden cambiar junto a otras tres compañeras por una (¡Una canica por tres, que desfachatez!) que recientemente haya conseguido grandes laureles en el campo de lucha del recreo del colegio.

Como se puede ver, la competencia es feroz, y rápidamente se pasa del éxito rotundo al fracaso más absoluto. Si hubiera algún modo de entrenarse en casa, ninguna otra canica se convertiría en rival de talla... pues Marble Madness nos proporciona esa posibilidad.

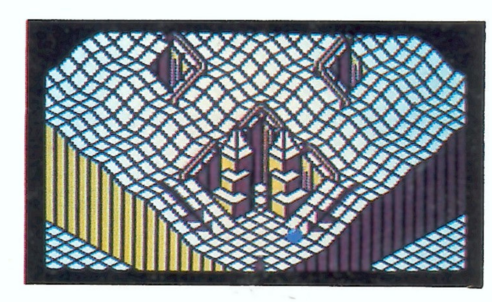
En este caso, de una realidad pasmosa, las carreras de canicas van a tener lugar en un espacio verdaderamente tridimensional, con subidas, bajadas, peligrosas curvas que pueden llevarnos al vacío, túneles... Podremos luchar contra el cronómetro en una carrera suicida o hacerlo contra otro compañero. Además, ésta no es una carrera normal, ya que aparecerán terribles enemigos a los cuales deberemos evitar o acabaremos con los «huesos» fuera de pista. Ni en nuestras más terribles pesadillas de canicas podríamos imaginarnoslos, son los mortales Steelies (ladronzuelos) o los contundentes martillos y sobre todos ellos, los más temibles son los Papeadores de canicas.

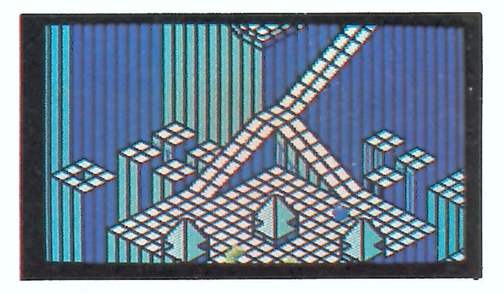
Cada nivel de pistas de carreras tiene personalidad propia y peligros diferentes, ¡y hay seis!, experimenta el placer de morir de un shock cardiaco en una verdadera carrera de... canicas.

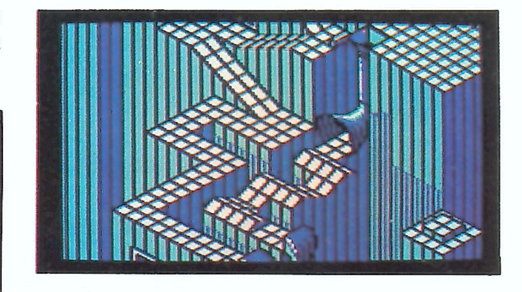


**VEREDICTO FINAL**

| | |
|---|---|
| GRAFICOS | ***** |
| SONIDO | *** |
| ORIGINALIDAD | *** |
| DIFICULTAD | Progresiva |
| INTERES | ***** |



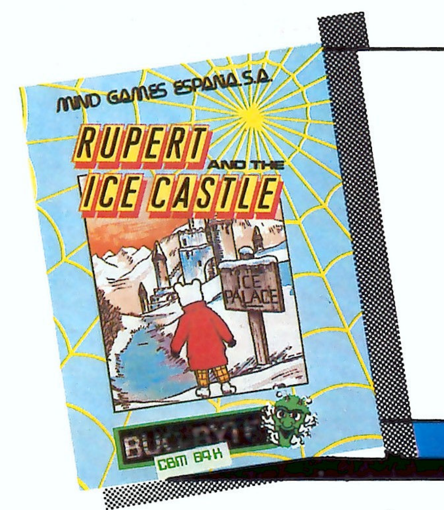






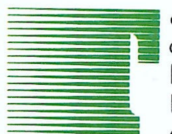
**FICHA TECNICA**

| | |
|---|---|
| NOMBRE | MARBLE MADNESS |
| PRECIO | 2.500 PTAS. |
| SOPORTE | DISCO |
| TIPO | COMPETICION |
| MODELO | C-64, C-128 |
| OBSERVACIONES | |




# Rupert y el castillo de hielo

¿Nunca habíais sentido el frio recorriendoos todo el cuerpo, introduciéndose por todas las rendijas de vuestras ropas y llegándoos hasta el tuétano? Pues ahora, expresiones tales como «me estoy congelando» o «me muero de frío» se van hacer realidad, si te cuentas entre los que estamos dispuestos a arriesgar nuestra vida entrando en el gélido mundo de Rupert y el Castillo de Hielo...



Todo empezó con una visita de cumplido por parte de los amigos de Rupert (Bingo, Edward Trunk, Algie y Badger Bill) a Jack el Escarchador, un buen compañero que habita el Castillo de Hielo. La tarde se iba desarrollando con la tranquilidad habitual, hasta que al acabarse la visita, y cuando los amigos de Rupert se iban ya, apareció Jenny, la traviesa hermana de Jack, no ocurriéndosele mejor diversión que congelarlos a todos.

Pero las noticias llegan pronto a Nutwood, y Rupert, enterado de las malas nuevas, ha decidido intentar rescatar a sus compañeros con nuestra ayuda y la de unas poderosas pastillas mágicas anticongelantes que se ha procurado. En cuanto logremos encontrar a nuestros amigos a lo largo de las estancias del Castillo de Hielo, podremos proporcionarles una de estas grageas, que les permitirán salir de su estado de congelación y volver sanos y salvos a Nutwood.

Claro que no todo se puede limitar a la facilidad de una búsqueda y Jenny lo sabe, por lo que ha decidido para su propia diversión colocarnos diversas trampas en el camino, tales como carámbanos de hielo que caen del techo, o cajas sorpresa, hombres de nieve, patines... que harán que perdamos cada vez una de nuestras pastillas, hasta el punto de que podemos acabar congelados del mismo modo si se nos agotan nuestras reservas.

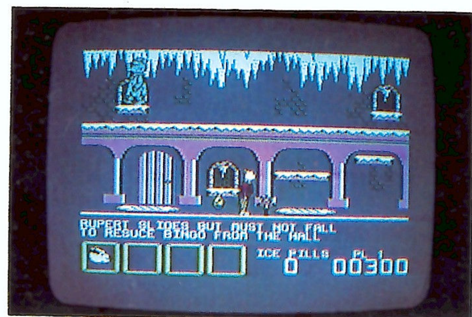
Guía a Rupert a través del palacio e impide que resbale y choque contra los innumerables peligros que le rodean, o acabaremos todos sirviendo de cubitos de hielo en las fiestas de Jenny.

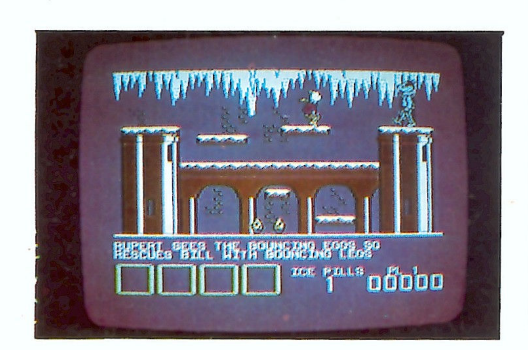


**VEREDICTO FINAL**

| | |
|---|---|
| GRAFICOS | **** |
| SONIDO | **** |
| ORIGINALIDAD | ** |
| DIFICULTAD | **** |
| INTERES | **** |

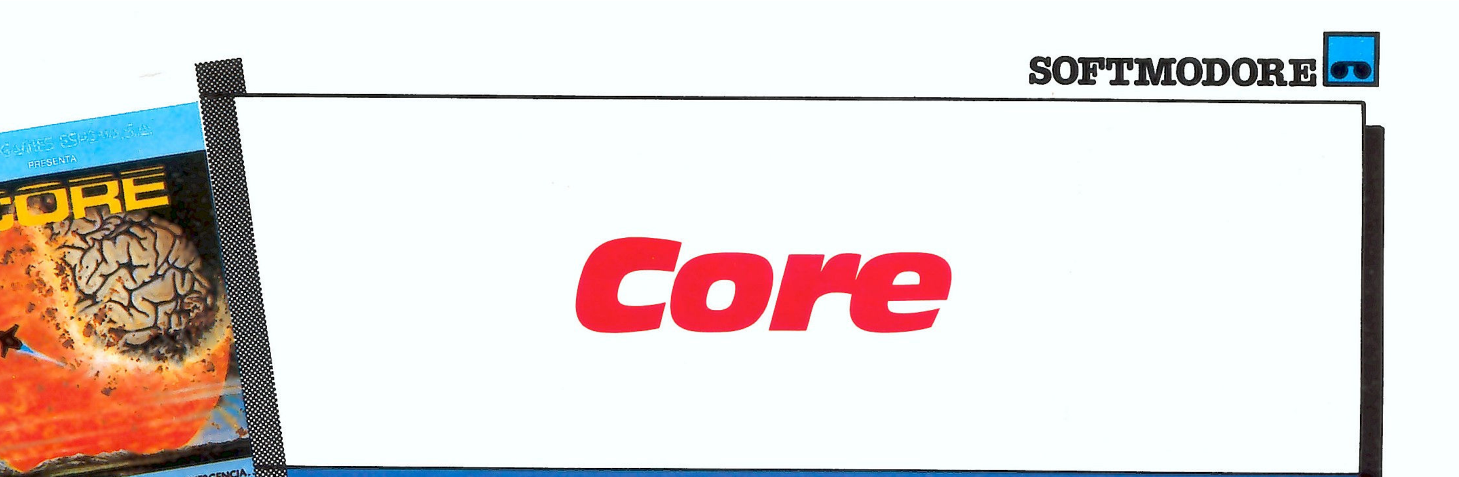



**FICHA TECNICA**

| | |
|---|---|
| NOMBRE | RUPERT Y EL CASTILLO DE HIELO. |
| PRECIO | 2.495 PTAS. |
| SOPORTE | CINTA |
| TIPO | ACCION |
| MODELO | C-64, C-128 |
| OBSERVACIONES | |




# Core

Cuando los viajes a través del espacio dejaron de ser un sueño inútil y el primer paso del hombre en la Luna se convirtió en mera anécdota, contada por los mayores a la luz de una chimenea fotoiónica, la alarma cundió en el espacio; por primera vez algo amenazaba el dominio de las estrellas. Si queremos saber de qué se trata, habremos de introducirnos en el increíble mundo de ... CORE.

Una vez estos viajes interestelares se hicieron una realidad palpable, nació la necesidad de encontrar un combustible apropiado para éstas gigantescas exploraciones de las galaxias. Tras múltiples investigaciones, se descubrió un fabuloso combustible químico, el cual sólo podía obtenerse en asteroides minerales. Se estableció el Proyecto de Explotación de Asteroides, llamado DAMP, con el objeto de desarrollar en esos satélites unas bases de explotación eficientes al máximo.

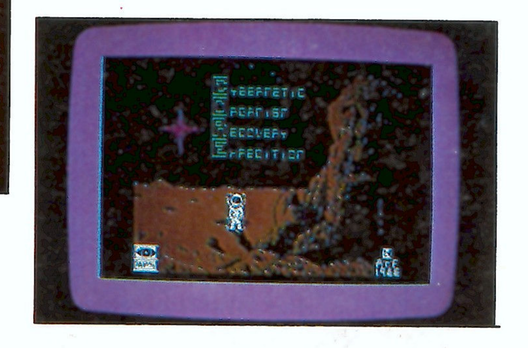
Los científicos, en su afán de estudio, descubrieron que en estos satélites anteriormente a la llegada humana había existido vida alienígena, y que por alguna razón, ésta desapareció para siempre... ¿o no?

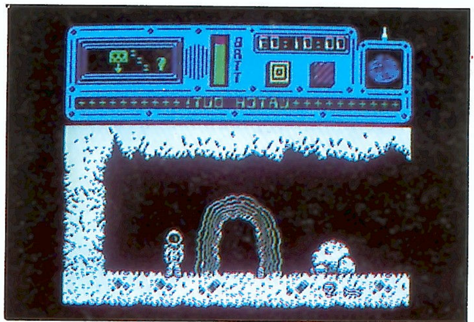
De pronto, el centro DAMP recibió una señal de urgencia, en la que claramente se avisaba de un peligro para toda vida humana presente en el planeta Eroc 1, perteneciente al proyecto y en el cual la explotación se encontraba en una fase algida. Por lo visto, este peligro consistía en un ataque alienígena y por ello DAMP decidió una arriesgada salida al problema.

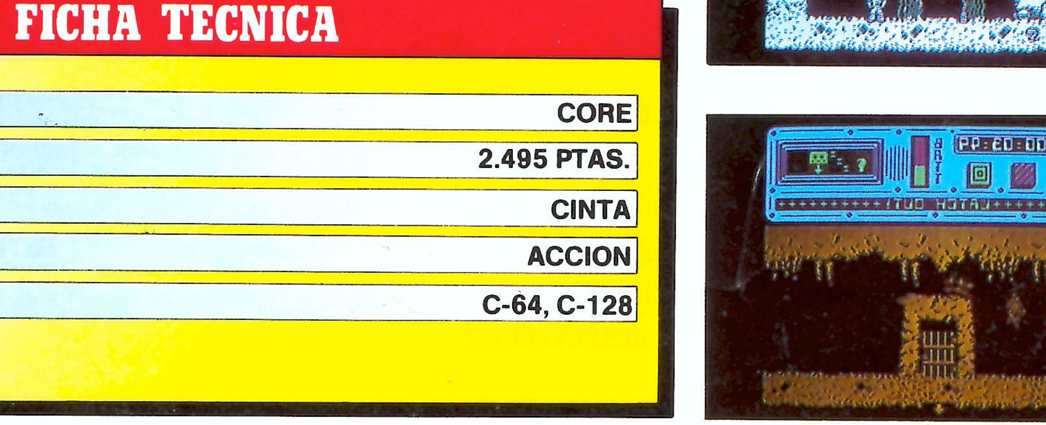
Seremos enviados a Eroc 1 para encontrar los segmentos de biomemoria que allí se hallan dispersados, estos segmentos son importantes porque permitirán elaborar una estrategia de defensa adecuada para repeler esos ataques alienígenas. El modo de viajar a través de Eroc 1 es por sus sistemas de telepuertas OTIM 351, ya que posée 4 niveles de más de 250 localizaciones celulares en cada uno. Así que, con mucha cautela (hay peligros a cada paso), deberemos enviar los segmentos de biomemoria a la Nave Nodriza y regresar sanos y salvos, aunque ésto... nos parece pedir demasiado.

## VEREDICTO FINAL

| | |
|---|---|
| GRAFICOS | **** |
| SONIDO | *** |
| ORIGINALIDAD | **** |
| DIFICULTAD | ***** |
| INTERES | *** |







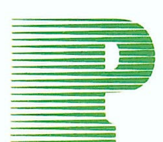
## FICHA TECNICA

| | |
|---|---|
| NOMBRE | CORE |
| PRECIO | 2.495 PTAS. |
| SOPORTE | CINTA |
| TIPO | ACCION |
| MODELO | C-64, C-128 |
| OBSERVACIONES | |

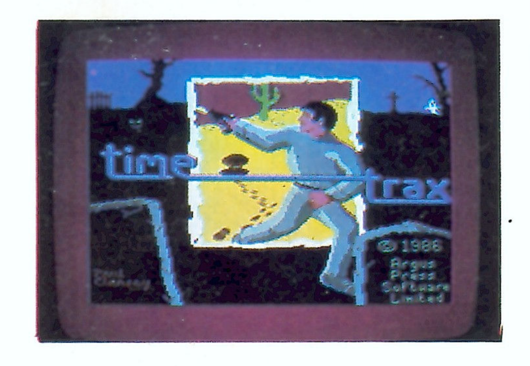



# Timetrax

Cuando las Ocho Mentes gobernaban el mundo, la vida transcurrída sosegada y plácidamente, hasta que cierto dia ocurrió lo más terrible, ni por los cerebros de esas psiques privilegiadas había pasado nunca la sospecha de tal desastre: sus Objetos de Carácter, las posesiones más apreciadas de las Ocho Mentes, fueron robadas y esparcidas a través de los tiempos...



Por ello, éstas mantuvieron abiertos los Portales del Tiempo, intentando hallarlos, y ahí empezó el verdadero problema. A través de estos portales abiertos se introdujeron las Fuerzas del Mal, muchas de las cuales han llegado para impedir que el mosaico del Universo Espacio-Tiempo pueda ser reconstruído y que retornen a sus primitivas manos los Objetos de Carácter, cerrando los Portales del Tiempo.




## VEREDICTO FINAL

| | |
|---|---|
| GRAFICOS | **** |
| SONIDO | ***** |
| ORIGINALIDAD | **** |
| DIFICULTAD | ***** |
| INTERES | ***** |

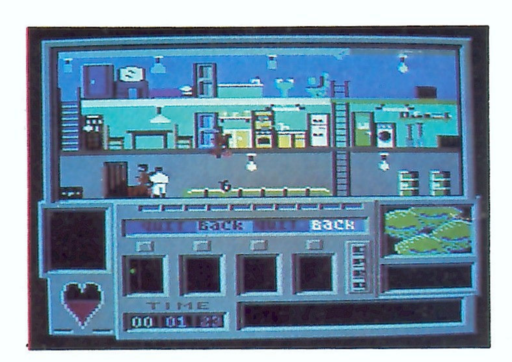
De este modo, nuestra misión va a consistir en rechazar a los Oscuros y recomponer el tejido del Universo, para lo cual necesitaremos encontrar los quince mosaicos que forman las Cuatro Runas Mágicas, las cuales permiten coser la tela del Tiempo-Espacio, y devolver a las Ocho Mentes sus Objetos de Carácter.

Puestos ya en antecedentes de nuestra complicada misión, iremos a través de las diferentes dimensiones temporales, defendiéndonos de los Oscuros que nos atacarán sin cesar, y buscando esos fragmentos de Universo tan preciados. Los Objetos de Carácter son ocho: navaja, libro, casco, muñeca de barro, calavera, mapa, escarabajo esmeralda y hacha de batalla, los cuales entregarás a las Mentes a medida que los vayas encontrando.

Podrás efectuar hechizos que te darán una fuerza especial, así como beber las pócimas que halles. Al final del juego, se te puntuará según tu éxito en la proporción del Diseño Cósmico descifrado, número de Objetos de Carácter devueltos a sus Mentes, cantidad de hechizos formulados y número de criaturas voladoras del lado Oscuro destruídas. Hay en total ocho zonas del Tiempo y multitud de Puertas Temporales ¿lograrás tu objetivo?

## FICHA TECNICA

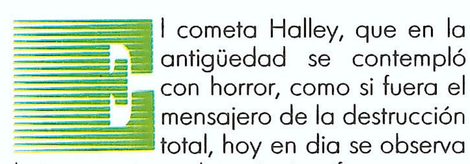
| | |
|---|---|
| NOMBRE | TIMETRAX |
| PRECIO | 2.495 PTAS. |
| SOPORTE | CINTA |
| TIPO | ESTRATEGIA Y ACCION |
| MODELO | C-64, C-129 |
| OBSERVACIONES | |






# The comet game

Multitud de misterios rodéan al Cosmos y todo lo que en él se encuentra. El hombre, para hallar nuestros ancestros y el orígen del mundo ha debido abrirse a la investigación del Universo, pero al mismo tiempo, ha sido necesaria una vigilancia del espacio exterior, nunca se sabe lo que puede existir vagando por las estrellas...



El cometa Halley, que en la antigüedad se contempló con horror, como si fuera el mensajero de la destrucción total, hoy en dia se observa bajo un matiz mucho más científico, una vez comprendidas esas apariciones periódicas. Como viajero incansable a través de nuestra galaxia, nos ha proporcionado abundante información, pero... ¿y si en verdad fuera una señal del holocausto?

Los científicos después de investigar arduamente su estructura, composición, longevidad, trayectoria... convergieron en un dramático descubrimiento: el cometa es capaz de albergar vida en forma de bacterias, pero lo peor es que éstas bacterias se suponen nocivas para cualquier forma de vida terrestre, por lo que un próximo paso del cometa por nuestra Tierra podría suponer la muerte total de todo ser vivo.

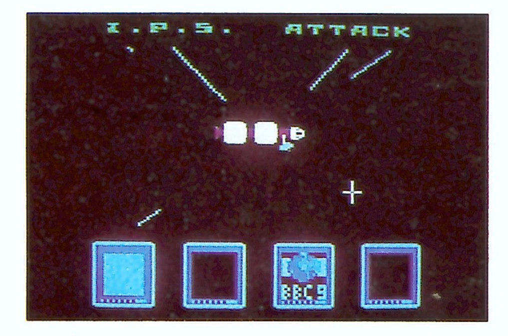
Hallado el gran problema, se intentó una posibilidad muy difícil, la de llegar hasta el cometa y rastrear la superficie en busca de grupos de bacterias letales, pero el viaje es largo y peligroso...

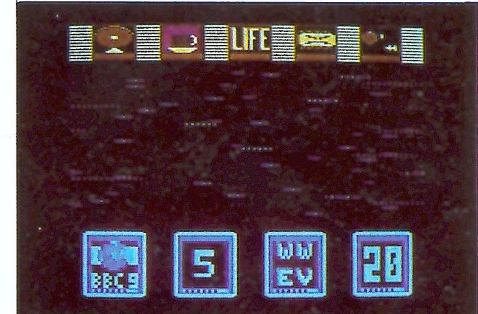
El piloto se encuentra en suspensión animada durante el vuelo, así, deberemos asumir el papel de computadora de a bordo, con cinco tareas relevantes: emparejar las secuencias aleatorias de LEDs (diodos emisores de luz) para conseguir que funcione un chip, defender la nave de un posible ataque interplanetario, sintonizar la antena en la frecuencia máxima, destruir posibles gérmenes en tu equipo de apoyo vital, y prepararle un buen café al piloto.

Una vez lleguemos al cometa, nos situaremos en órbita baja sobre su superficie. A partir de aquí, deberemos disparar contra los grupos de bacterias tóxicas, destruyéndolas lo más rápido posible, ya que la Tierra depende de nosotros.

## VEREDICTO FINAL

| | |
|---|---|
| GRAFICOS | *** |
| SONIDO | *** |
| ORIGINALIDAD | ***** |
| DIFICULTAD | ***** |
| INTÉRES | *** |



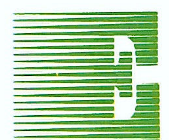




## FICHA TECNICA

| | |
|---|---|
| NOMBRE | THE COMET GAME |
| PRECIO | 2.200 PTAS. |
| SOPORTE | CINTA |
| TIPO | ACCION |
| MODELO | C-64, C-128 |
| OBSERVACIONES | |

# Espría

En muchas ocasiones, hemos podido ver (y tal vez admirar) los maravillosos sprites que pueblan los programas más conocidos. Y probablemente, habremos deseado que hubiese alguna forma de utilizar uno o varios de ellos en nuestros programas. Si es así, ESPRIA es el programa: permitirá espiar a los sprites...



El programa que hoy presentamos tiene una misión muy específica: «explorar» la memoria del ordenador, volcando su contenido en un sprite que tenemos a la vista. De esta forma, todos los sprites almacenados en la memoria del ordenador pueden ser visualizados sin esfuerzo.

Pero eso no es todo: Además de verlos, podemos almacenar los sprites que nos gusten en cinta o disco, para su uso posterior. Y mediante el segundo de los listados, crear las líneas DATA que definen dichos sprites, para poder insertarlos directamente en nuestros programas BASIC.

## *Una buena pregunta...*

Os preguntaréis, no sin razón, que de dónde van a salir esos sprites en la memoria del 64, puesto que la aparición espontánea no es posible (para desgracia de los programadores, ¿verdad?).

En general, ESPRIA permite ver los sprites que previamente han definido otros programas en la memoria; por ejemplo, un juego previamente cargado en el ordenador.

Para poder utilizar nuestro «007 Commodoriano, con licencia para piratear», necesitamos pues, no sólo cargar el programa cuyos sprites deseamos copiar, sino salir de él sin borrar la memoria del ordenador (y con ella a los sprites...). Es decir, hay que encontrar una forma de inicializar el ordenador: es decir, RESETEAR, y que la Real Academia de la Lengua nos perdone, porque la palabrita ni «limpia», ni «brilla», ni «da esplendor», pero nos lo ha dejado muy claro, porque por suerte o por desgracia todos la hemos comprendido.

Con los programas en BASIC, o ciertos programas en código máquina, la solución es sencilla: pulsando la tecla RUN/STOP detendremos el programa, permitiéndonos cargar ESPRIA y seguir la operación.

En la mayoria de programas comerciales, sin embargo, la cosa no es tan sencilla. Algunos de ellos poseen una opción de abandonar el juego al final, volviendo al BASIC; no obstante, estos programas son minoría. Para gran parte de los programas comerciales, será preciso emplear un botón de reset.

El Commodore 64 no se halla dotado de un botón de reset, por lo que deberemos pasar a la construcción de uno. La descripción del reset podemos encontrarla en los números 4 y 5 de TU MICRO COMMODORE, a los que os remitimos para más detalles sobre este asunto.

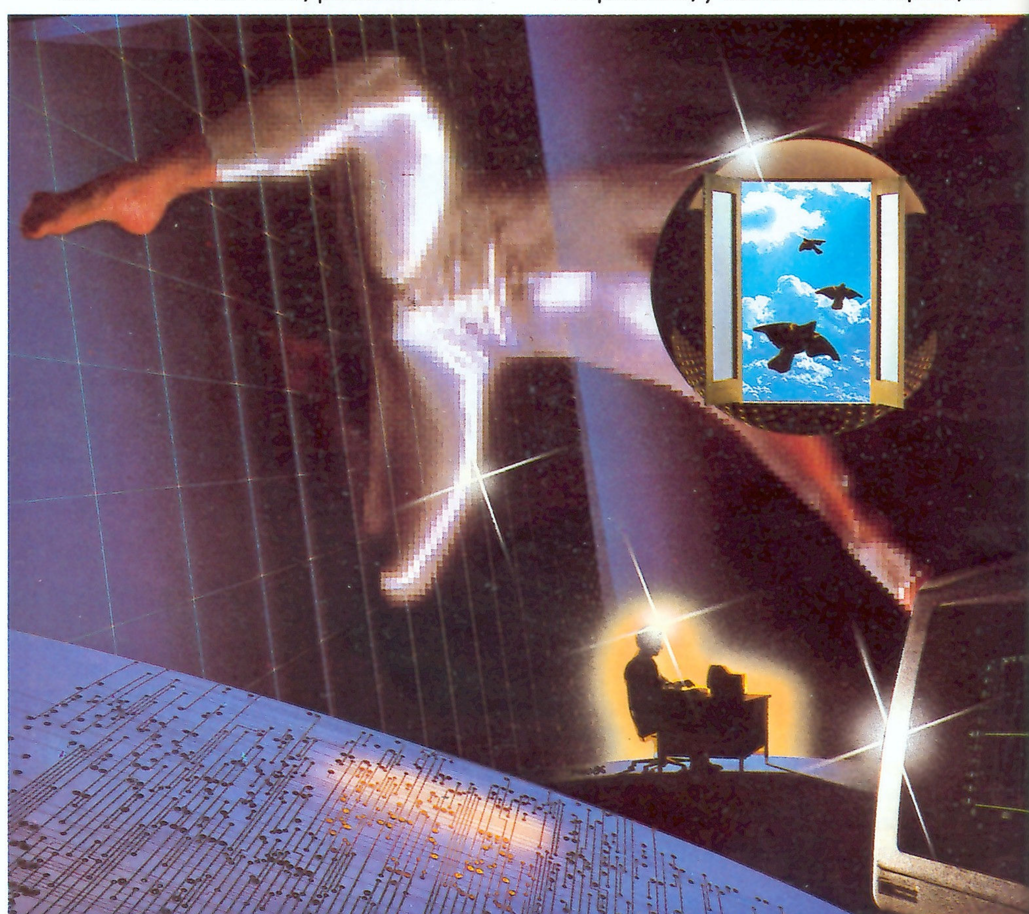
Con un botón de reset, podemos acceder a bastantes programas, para descubrir sus sprites más escondidos. Desgraciadamente, algunos programas comerciales se protegen bloqueando el botón de reset, por lo que a éstos no podremos «espriarlos», si no es empleando métodos excesivamente sofisticados...

Y hablando del tema protecciones: recordad que, en todo caso, el diseño de los sprites que pudiéseis copiar de cualquier programa sigue siendo propiedad del autor del mismo, así que cuidado con los copyrights...

## *Y ahora señores, pasen y vean... ¡sprites!*

Vamos a pasar a describir el programa. En primer lugar, podréis ver el listado del programa SALVAESPRIA. Este programa debe ser tecleado y grabado antes de su uso.

Al ejecutarlo, y tras una breve espera, se



nos preguntará si deseamos grabar ESPRIA en disco o cinta. Pulsando D o C elegiremos el periférico adecuado a nuestra configuración hardware.

Una vez hecho esto, se grabará la rutina ESPRIA en el medio deseado. Tened en cuenta que, a la hora de grabar los sprites, el SPRIA de disco los grabará directamente a disco, y el de cinta a cinta, por lo que no es posible el uso combinado de ambos periféricos.

El programa ESPRIA que debemos utilizar es el que graba el programa anterior. Para ello, sigamos las instrucciones dadas por SALVAESPRIA: según se trate de cinta o disco, deberemos ejecutar un LOAD adecuado, introducir a continuación el comando NEW y posteriormente: SYS 49152.

En realidad, SALVAESPRIA crea, en el disco o el cassette, una copia de un corto programa en código máquina, que es el realmente útil. Los comandos descritos en el párrafo anterior lo cargan y ejecutan.

Y aquí nos encontramos ya, utilizando el único, el inigualable, el auténtico... ESPRIA.

En la pantalla se verá un rótulo de presentación, y bajo él el mensaje «POSICION» seguido de un número, que muestra la dirección de la memoria que está siendo examinada.

Pulsando las teclas «+» y «—» podemos, respectivamente, avanzar o retroceder 64 posiciones de memoria, es decir, examinar el sprite anterior o posterior. Mediante las teclas «X» e «Y» se amplía el sprite en los sentidos horizontal y vertical, respectivamente. Una segunda pulsación de la tecla devuelve el sprite a su estado normal.

Las teclas de función cumplen un cometido especial: mediante F1 podemos conmutar entre sí el color de sprite entre los modos de color simple o multicolor.

Con las teclas F3, F5 y F7 se van cambiando los colores del sprite, tanto el del propio sprite como los colores auxiliares del modo multicolor.

Con todas estas opciones, es posible cambiar el aspecto y color del sprite, hasta observar su forma adecuada.

La tecla «D» vuelca los datos del sprite que se está visualizando al periférico indicado.

Un aviso: dado que el programa ha de examinar la memoria, debe ser lo más corto posible, para reducir la posibilidad de sobreescribir algunos sprites que, casualmente, ocupen la misma zona de memoria. Por tanto, la rutina ha debido simplificarse al máximo.

La simplificación se ha llevado a cabo a costa de no incluir rutinas de proceso de errores; así pues, si vamos a grabar en disco, asegurémonos que el soporte esté correctamente insertado, o si es en cinta, que se han pulsado las teclas RECORD y PLAY. Un error no será detectado, y tal vez, pueda producir la pérdida de los datos que estamos grabando.

Finalmente, la tecla «←» (flecha atrás) cierra el fichero de datos y vuelve al BASIC.

## Y además... ¡Generador de «DATAS»!

El segundo listado es un ejemplo de cómo podemos aprovechar los ficheros creados por ESPRIA. En este caso, el programa crea las líneas DATA que reproducen fielmente la forma de cada sprite. Teclead el programa y grabadlo para su uso posterior. No lo ejecutéis sin grabarlo previamente.

Cargado y ejecutado el programa, se nos pedirá el periférico que contiene el fichero de datos (que, por cierto, se llama SPRITE, para los que quieran manipularlo por su cuenta).

Una vez indicado, nos pedirá que esperemos mientras procede a las operaciones necesarias. Finalmente, volverá al BASIC. Si ahora listamos el programa, podremos ver que se han generado, por arte de magia informática, las líneas DATA a partir del número 1000. Borrando el programa generador, tendremos dichas líneas para su uso.

Deseamos que paséis un buen rato con este programa, y que los sprites mágicos se salgan de sus juegos de siempre y se metan en los vuestros...

## LISTADO

LISTADO SALVAESPRIA

```
1 REM ****************                             -245-
2 REM * SALVA-ESPRIA *                             -082-
3 REM ****************                             -247-
10 POKE53280,.:POKE53281,.                         -210-
20 PRINT"{CLR}{BLK}";:                             -235-
30 PRINT"{CLR}[8 ESP]{PUR}SINDIN SOFT PRESENTA[3 .]":
PRINT                                              -119-
40 PRINT"[26 ESP]{RON}[6 ESP]{ROF}[8 ESP]";        -052-
50 PRINT"[25 ESP]{RON} {ROF}[6 ESP]{RON} {ROF}[7 ESP]
";                                                 -217-
60 PRINT"[24 ESP]{RON} {ROF}[3 ESP]{RON}<B><I>{ROF}<B
>[2 ESP]{RON} {ROF}[6 ESP]";                       -062-
70 PRINT"[23 ESP]{RON} {ROF}[3 ESP]{RON}<B>{ROF}[3 ES
P]<B>[2 ESP]{RON} {ROF}[5 ESP]";                   -189-
80 PRINT"[8 ESP]<A>¦*¦<S><A>¦*¦<S><A>¦*¦<S><A>¦*¦<S><
R> {RON} {ROF}[4 ESP]<K>[3 ESP]{RON}<K>{ROF}[3 ESP]{R
ON} {ROF}[4 ESP]";                                 -012-
90 PRINT"[8 ESP]¦-¦[2 ESP]¦-¦[2 ESP]¦-¦ ¦2 -¦ ¦2 -¦ {
RON} {ROF}[4 ESP]<K>[3 ESP]{RON}<K>{ROF}[3 ESP]{RON}
{ROF}[4 ESP]";                                     -174-
100 PRINT"[8 ESP]<Q><W> <Z>¦*¦<S><Q>¦*¦<X><Q><R><X>¦-
¦ {RON} {ROF}[4 ESP]{RON}<D><3 I><F>{ROF}[3 ESP]{RON}
 {ROF}[4 ESP]";                                    -110-
```

```
110 PRINT"[8 ESP]¦-¦[4 ESP]¦2 -¦[2 ESP]¦-¦<Z><S>¦-¦ {
RON} {ROF}[4 ESP]<K>[3 ESP]{RON}<K>{ROF}[3 ESP]{RON}
{ROF}[4 ESP]";                                   -120-
120 PRINT"[8 ESP]<Z>¦*¦<2 E>¦*¦<2 E>¦2 *¦<E> <2 E> {R
ON} {ROF}[4 ESP]<K>[3 ESP]{RON}<K>{ROF}[3 ESP]{RON} {
ROF}[4 ESP]";                                    -177-
130 PRINT"[23 ESP]{RON} {ROF}[3 ESP]<K>[3 ESP]{RON}<K
>{ROF}[2 ESP]{RON} {ROF}[5 ESP]";                -174-
140 PRINT"[23 ESP]{RON}[2 ESP]{ROF} <C>{RON}<I>{ROF}[
3 ESP]{RON}<I>{ROF}<V>{RON} {ROF}[6 ESP]";       -143-
150 PRINT"[22 ESP]{RON}[4 ESP]{ROF}[6 ESP]{RON} {ROF}
[7 ESP]";                                        -010-
160 PRINT"[21 ESP]{RON}[3 ESP]{ROF}[2 ESP]{RON}[6 ESP
]{ROF}[8 ESP]";                                  -139-
170 PRINT"[20 ESP]{RON}[3 ESP]{ROF}[17 ESP]";
                                                 -104-
180 PRINT"[19 ESP]{RON}[3 ESP]{ROF}[18 ESP]";
                                                 -105-
190 PRINT"[18 ESP]{RON}[3 ESP]{ROF}[19 ESP]";
                                                 -106-
200 PRINT"[17 ESP]{RON}[3 ESP]{ROF}[20 ESP]";
                                                 -098-
210 PRINT"[17 ESP]{RON}[2 ESP]{ROF}[21 ESP]"
                                                 -040-
220 PRINT"[8 ESP]{PUR}(C) 1986 TU MICRO COMMODORE"
                                                 -213-
230 PRINT"{ABJ}[13 ESP]ESPERE, POR FAVOR[3 .]"
                                                 -096-
240 FORI=49152TO49558:READA:POKEI,A:NEXT:A=780:X=781:
Y=782                                            -064-
250 PRINT"{HOM}{23 ABJ}{12 DCH}[3 ESP]{RON}D{ROF}ISCO
 O {RON}C{ROF}INTA ?[4 ESP]"                     -117-
260 GETP$:IFP$<>"D"ANDP$<>"C"THEN260             -090-
270 IN=49152:AI=INT(IN/256):BI=IN-AI*256         -073-
280 FI=49562:FI=FI+1:AF=INT(FI/256):BF=FI-AF*256
                                                 -031-
290 P=8:IFP$="C"THENP=1:POKE49427,1:POKE49418,144:POK
E49420,193                                       -171-
300 N$="ESPRIA":FORI=1TOLEN(N$):POKE10751+I,ASC(MID$(
N$,I,1)):NEXT                                    -194-
310 POKE251,BI:POKE252,AI:D=255                  -065-
320 POKEA,1:POKEX,P:POKEY,D:SYS65466             -053-
330 POKEA,LEN(N$):POKEX,0:POKEY,42:SYS65469      -172-
340 POKEA,251:POKEX,BF:POKEY,AF:SYS65496         -028-
350 PRINT"{CLR}PROGRAMA 'ESPRIA' GRABADO EN ";
                                                 -197-
360 IF P=1THENPRINT"CINTA"                       -149-
370 IF P=8THENPRINT"DISCO"                       -160-
380 PRINT"{2 ABJ}PARA USARLO: {RON}LOAD'ESPRIA',"P"{I
ZQ},1"                                           -032-
390 PRINT"{ABJ}[13 ESP]{RON}NEW":PRINT"{ABJ}[13 ESP]{
RON}SYS 49152"                                   -044-
400 DATA 169,0,141,32,208,141,33,208,32,68,229,160,19
3,169,62,32                                      -142-
410 DATA 30,171,169,170,141,0,208,169,100,141,1,208,1
69,0,141,16                                      -118-
420 DATA 208,169,13,141,248,7,169,0,141,28,208,169,5,
141,39,208                                       -104-
430 DATA 169,7,141,37,208,169,2,141,38,208,169,1,141,
21,208,169                                       -103-
440 DATA 128,141,138,2,162,3,160,19,24,32,10,229,165,
254,166,253                                      -137-
450 DATA 32,205,189,160,193,169,121,32,30,171,120,169
,0,133,1,160                                     -176-
460 DATA 0,177,253,153,64,3,200,192,64,208,246,169,7,
133,1,88                                         -004-
470 DATA 32,228,255,201,43,208,14,165,253,24,105,64,1
33,253,144,3                                     -184-
480 DATA 230,254,76,68,192,201,45,208,14,165,253,56,2
33,64,133,253                                    -000-
490 DATA 176,3,198,254,76,68,192,201,133,208,11,173,2
8,208,73,1                                       -112-
500 DATA 141,28,208,76,68,192,201,134,208,6,238,37,20
8,76,68,192                                      -167-
510 DATA 201,135,208,6,238,38,208,76,68,192,201,136,2
08,6,238,39                                      -159-
520 DATA 208,76,68,192,201,95,208,14,169,0,141,21,208
,169,2,32                                        -048-
530 DATA 195,255,32,68,229,96,201,88,208,11,173,29,20
8,73,1,141                                       -112-
540 DATA 29,208,76,68,192,201,89,208,11,173,23,208,73
,1,141,23                                        -051-
```

```
550 DATA 208,76,68,192,201,68,240,3,76,68,192,173,143
,193,201,0                                          -109-
560 DATA 208,26,169,1,141,143,193,169,13,162,130,160,
193,32,189,255                                      -047-
570 DATA 169,2,162,8,160,2,32,186,255,32,192,255,162,
2,32,201                                            -246-
580 DATA 255,160,0,140,21,208,185,64,3,32,210,255,200
,192,64,208                                         -129-
590 DATA 245,169,1,141,21,208,169,3,32,201,255,76,68,
192,32,32                                           -051-
600 DATA 32,32,32,69,83,80,82,73,65,32,40,67,41,32,84
,85                                                 -013-
610 DATA 32,77,73,67,82,79,32,67,79,77,77,79,68,79,82
,69                                                 -078-
620 DATA 13,13,13,32,32,32,32,32,32,32,32,32,32,80,79
,83                                                 -231-
630 DATA 73,67,73,79,78,32,61,32,0,32,32,32,32,32,32,
32                                                  -203-
640 DATA 32,0,64,48,58,83,80,82,73,84,69,44,83,44,87,
0                                                   -188-
650 DATA 83,80,82,73,84,69,0,0,0                    -103-

READY.                                             -173-
```

LISTADO GENERADOR DE DATAS ESPRIA

```
1 REM GENERADOR DE DATAS                            -034-
2 POKE53280,.:POKE53281,.:PRINT"{CLR}{WHT}"         -070-
3 PRINT"{HOM}{13 ABJ}{13 DCH}{RON}D{ROF}ISCO O {RON}C
{ROF}INTA ?"                                        -164-
4 GET A$:IFA$<>"C"ANDA$<>"D"THEN4                   -133-
5 PRINT"{CLR}[11 ESP]ESPERE, POR FAVOR[3 .]"
                                                    -066-
6 IF A$="C"THENOPEN1,1,0,"SPRITE":GOTO8             -107-
7 OPEN1,8,2,"SPRITE,S,R"                            -192-
8 ME=49152                                          -044-
9 FORI=.TO63                                        -000-
10 GET#1,A$:POKEME+I,ASC(A$+CHR$(0))                -027-
11 NEXT                                             -193-
12 IF(ST)AND64THENCO=CO+2:CLOSE1:GOTO14             -237-
13 ME=ME+64:CO=CO+1:GOTO9                           -029-
14 POKE254,CO                                       -013-
15 POKE254,PEEK(254)-1:IFPEEK(254)=0THENPRINT"{CLR}{W
HT}";:POKE198,.:END                                 -123-
16 NL=1000                                          -031-
17 PM=49152                                         -103-
18 PRINT"{CLR}{BLK}{2 ABJ}";                        -218-
19 FORI=.TO3                                        -251-
20 PRINTNL"DATA";:NL=NL+10                          -121-
21 FORJ=.TO15                                       -040-
22 A=PEEK(PM+16*I+J):A$=STR$(A)                     -026-
23 PRINTRIGHT$(A$,LEN(A$)-1)",";:NEXT               -173-
24 PRINT"{IZQ} ":NEXT                               -141-
25 PRINT"16 NL="NL                                  -080-
26 PRINT"17 PM="PM+64                               -237-
27 PRINT"GOTO 15"                                   -025-
28 POKE198,9:FORI=.TO9:POKE631+I,13:NEXT            -241-
29 PRINT"{HOM}";:END                                -187-

READY.                                             -173-
```

# Chip de vídeo VIC II

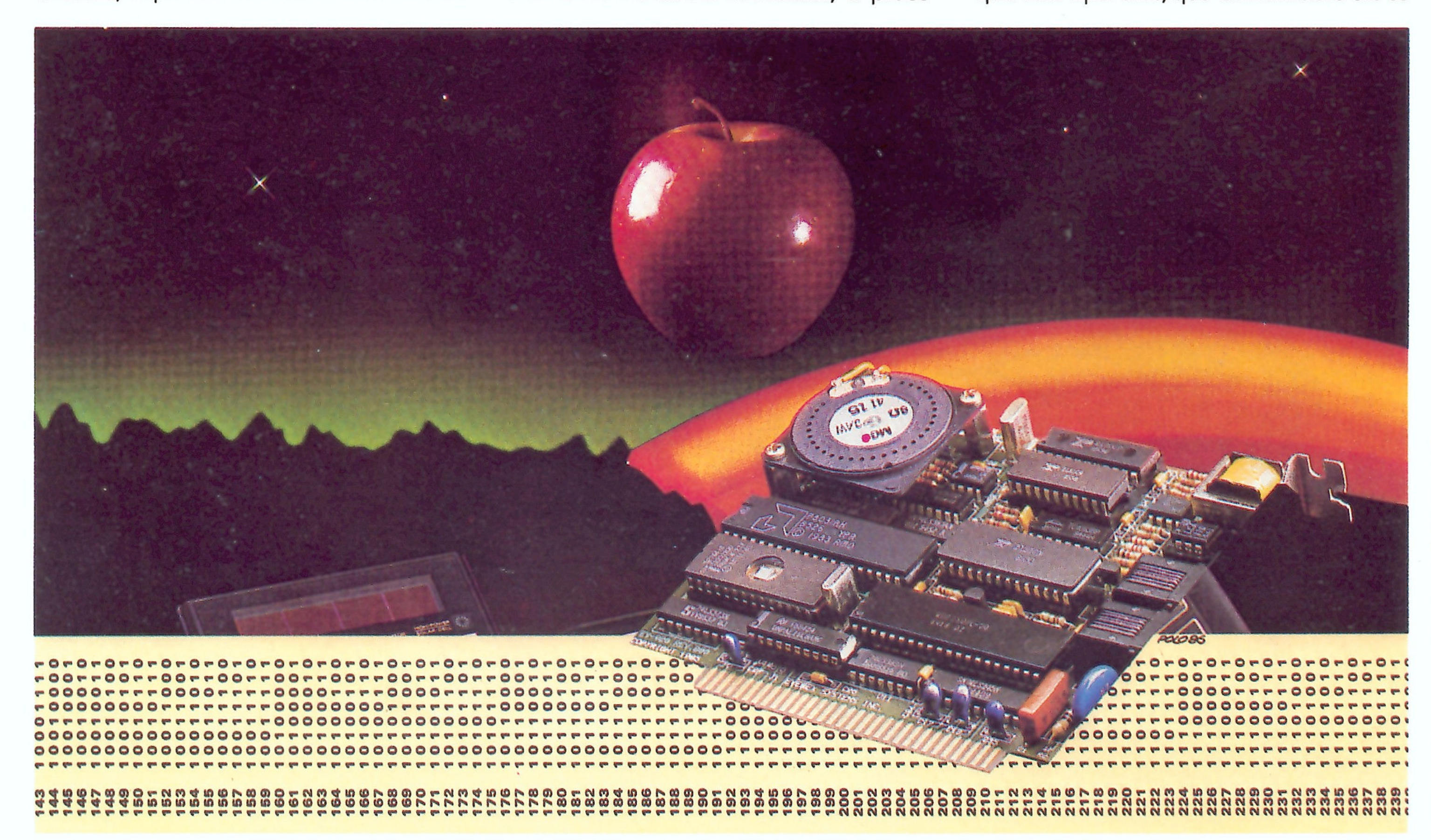
Programar en código máquina tiene sus ventajas; por ejemplo, nos permite acceder directamente a los registros de todos los dispositivos internos del ordenador. Pero para ésto, no basta conocer el microprocesador y sus instrucciones; es necesario, además, saber cual es la función de cada registro determinado.

Efectivamente, la programación en código máquina es mucho más que conocer y manejar el juego de instrucciones del microprocesador. Para realizar cualquier tarea compleja con eficacia, es preciso conocer lo más a fondo posible todos los dispositivos de la máquina, sus funciones y como éstas pueden ser accedidas por el microprocesador para realizar unas tareas determinadas.

Afortunadamente, están ya lejos los tiempos en que los dispositivos del ordenador eran unos monstruos intocables; el procesador debía adaptarse a ellos y sus limitaciones, en lugar de ser al contrario. En la actualidad, la inmensa mayoría de los dispositivos internos son programables; es decir, el microprocesador puede acceder a ellos y «decirles» como quiere que se comporten.

Lo que hasta ahora hemos llamado «dispositivos» suelen ser circuitos integrados, especializados en una tarea determinada, como generar sonido, controlar las comunicaciones, los periféricos, etc.

Vamos a estudiar hoy uno de los chips más potentes de Commodore: el Video Interface Controller (VIC, para los amigos). Tan importante es su función para nuestros queridos aparatos, que Commodore dió su



nombre a uno de los primeros equipos: el ya famoso (y desgraciadamente en vías de extinción) VIC 20. Este chip es el encargado de gestionar todas las tareas de generación de gráficos: caracteres, alta resolución y sprites son sus habilidades, nada despreciables por cierto...

Pero antes de estudiar al amigo VIC, puede resultar interesante ver cómo se genera una imagen en nuestro televisor o monitor. Una descripción exhaustiva excedería los límites de espacio de que disponemos; nos contentaremos, pues, con tratar los principios básicos del televisor.

## La magia de una imagen electrónica

La siguiente afirmación, tal vez sorprenda a muchos de vosotros: el televisor no crea imágenes. Todo lo que vemos es un punto en rapidísimo movimiento.

Dicho punto, al cual llamaremos raster (no le perdáis de vista, pues permite efectos sorprendentes desde el VIC), es un pequeño círculo luminoso, cuya luminosidad varía. La imagen recogida por una cámara es descompuesta en un gran número de puntos, de distinta luminosidad y color cada uno. Dichos puntos se transmiten después en sucesión, uno tras otro, y de forma ordenada.

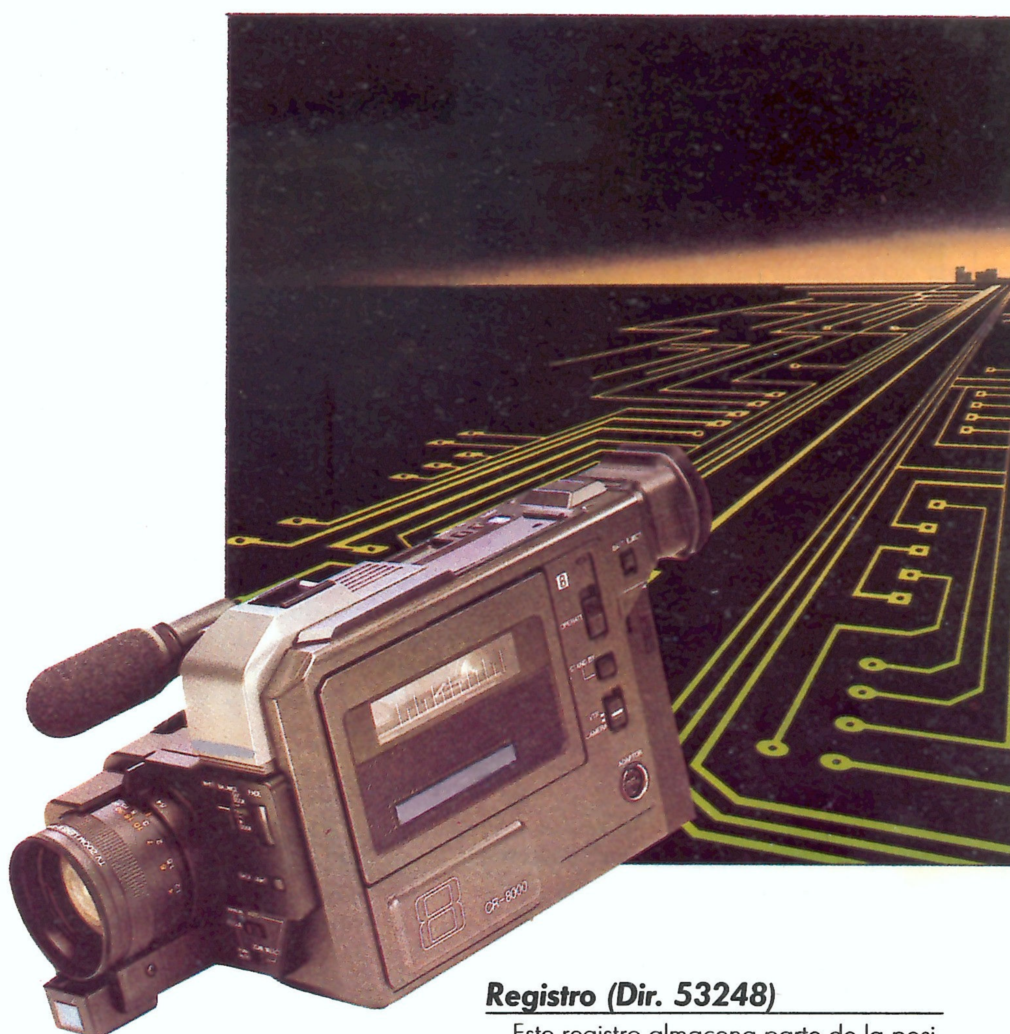
Mientras, el raster barre la pantalla del receptor. El barrido comienza en la esquina superior izquierda de la pantalla y el rayo luminoso se mueve de izquierda a derecha, a través de la misma; su intensidad luminosa y color depende de la señal que le llegue en ese instante desde el transmisor, con lo cual se reproduce una línea de la imagen deseada. Al llegar al extremo derecho de la pantalla, baja un poco, vuelve al extremo izquierdo y repite la operación, bajando un poco más cada vez, hasta llegar a la parte inferior de la pantalla, desde donde sube otra vez y repite el ciclo.

Una pantalla completa se compone de unas 500 líneas (teóricamente 625, pero algunas se pierden fuera de la pantalla), y se denomina un cuadro. Se transmiten 50 cuadros por segundo, necesarios para dar sensación de una imagen contínua y en movimiento.

## Registros: posiciones de memoria... con vistas al CHIP

Tras estos preliminares, vamos a estudiar los registros del VIC y cada una de sus funciones; pero... ¿Qué es un registro?

A grandes rasgos, un registro no es más que una posición de memoria, desde el punto de vista del microprocesador. Pero es un byte que pertenece, no a la RAM, sino al chip en cuestión; en esta posición, cada uno de los bits controla una función o un modo de trabajo del dispositivo, o bien contiene un dato que el dispositivo devuelve al procesador.



El símil más obvio es el de una caja con un fondo móvil de cristal; el microprocesador deposita en ella un valor, como hace con todas las demás cajas de la memoria (con STA). Sin embargo, a través del fondo de cristal, el chip ve lo que hay en ella, y actúa en consecuencia. Cuando quiere devolver un dato al microprocesador, abre el fondo y lo deposita en la caja, con lo que queda listo para ser accedido por la CPU (con un simple LDA, por ejemplo).

Visto esto, pasemos revista a los registros del VIC II:

La dirección base de los registros es $D000, lo cual significa que, siendo los registros como posiciones de memoria consecutivas, la dirección de la primera de ellas en la memoria es $D000, es decir, 53248. Por tanto, la dirección del registro 14 será 53248 + 14 = 53262; a esta posición deberemos efectuar un POKE en código máquina, (instrucción STA), o leer (con LDA).

## Registro (Dir. 53248)

Este registro almacena parte de la posición X del primer sprite (sprite 0). Decimos parte, pues cada sprite tiene una posición X de 0 a 511, por lo que es necesario un bit adicional (recordemos que, con 8 bits, podemos generar de 0 a 255). Este se halla en otro registro que veremos más adelante, que es accesible en lectura y escritura, al igual que todos los demás del VIC.

## Registro 1 (Dir. 53249)

Se almacena aquí la posición y del sprite 0. Esta posición y puede estar comprendida entre 0 y 255, aunque las visibles están entre 50 y 250. Valores fuera de estos límites mueven el sprite bajo el marco de la pantalla, donde no son visibles. No obstante, ésto proporciona un bonito efecto de «esfumado y mutis por el foro» del sprite.

## Registros 2, 4, 6, 8, 10, 12, 14

Contienen las posiciones X de cada uno de los sprites, del 1 al 7. Su descripción es la misma que para el registro 0, por lo que no os cansaremos repitiendo 7 veces más la misma historia.

## Registros 3, 5, 7, 9, 11, 13, 15

Almacenan la posición y para cada uno de los restantes sprites.

## Registro 16 (Dir. 53264)

Este es el registro que contiene el bit más significativo de la posición X de cada sprite. El bit 0 pertenece al sprite 0, el 1 al uno y así para los ocho. Siempre que dispongamos de un programa para mover sprites, debemos prever que, al exceder de 255, se ponga a uno este bit, y viceversa.

## Registro 17 (Dir. 53265)

En este caso, cada bit tiene una función diferente, y controla un modo de trabajo del VIC:

— BIT 0, 1, 2: Controlan la posición vertical de la pantalla (de 0 a 7). En efecto, el VIC permite mover la pantalla completa, en dirección vertical, a saltos de un pixel (punto de la pantalla). Esta característica se emplea para realizar un scroll suave de la pantalla en dirección vertical, junto a otras características que veremos más adelante.

—BIT 3: Si se halla a uno, la pantalla dispone de 25 líneas. Al pasar a 0, la línea superior queda oculta por el marco de la pantalla, dejando visibles sólo 24 líneas. Esta función se emplea durante el scroll, proporcionando un lugar no visible donde colocar los nuevos caracteres que han de moverse por la pantalla.

— BIT 4: Este bit desactiva la pantalla. Cuando hayáis cargado un programa desde cinta, habréis observado que la pantalla desaparece. Esto se hace poniendo el bit a 0. De este modo, el VIC deja libre el bus de microprocesador, aumentando la velocidad del mismo en un 15%.

— BIT 5: Este bit selecciona el modo texto (0) o alta resolución (1).

— BIT 6: Se controla con este bit el modo de color extendido (1) o color normal (0).

El modo de color extendido permite representar cada carácter con un color de fondo distinto: por ejemplo, una A blanca, sobre fondo negro en una pantalla azul. En este modo, sólo es posible representar los 64 primeros caracteres de la tabla; los dos bits más significativos del código del carácter (el número que se introduce en la RAM de pantalla para representar el carácter) se interpretan como indicativos del registro en el que buscar el color de fondo, según la siguiente tabla:

00 — Registro color fondo 0
01 — Registro color fondo 1
10 — Registro color fondo 2
11 — Registro color fondo 3

— BIT 7: Es el bit más significativo de la posición vertical del raster (¿lo recordáis?) en la pantalla; esta posición está entre 0 y 511. Los 8 bits menos significativos se agrupan en el registro siguiente.

## Registro 18 (Dir. 53266)

Contiene los 8 bits menos significativos de la posición del raster. Este registro puede leerse y escribirse. En lectura, contiene siempre el valor de la posición del raster. Si escribimos en el registro, el valor actual de la posición de raster (veremos el significado exacto de ésto al hablar de las interrupciones por barrido).

## Registro 19 (Dir. 53267)

En este registro se halla la posición horizontal del lápiz óptico en la pantalla (obviamente, si disponemos de uno; de lo contrario, el valor contenido no será significativo). Dado que la pantalla tiene 320 pixel de ancho, sería necesario un bit adicional. Al no haberse previsto el mismo, la resolución que se obtiene será algo mayor de un pixel.

## Registro 20 (Dir. 53268)

Contiene la situación vertical del lápiz óptico.

## Registro 21 (Dir. 53269)

Cada uno de los bits de este registro controla la activación de los sprites: el bit 0 controla si el sprite 0 está visible (1) o invisible (0), así para cada sprite. Obviamente, para que el sprite sea visible debe estar dentro de la pantalla.

## Registro 22 (Dir. 53270)

Este es nuevamente un registro en el cual cada bit controla una función diferente del VIC:

— BIT 0, 1, 2: Contienen la posición horizontal de la pantalla (de 0 a 7). Esta característica permite desplazar la pantalla completa pixel a pixel, en cualquier sentido. Nos permite realizar un scroll suave, como veremos próximamente.

— BIT 3: Permite que la columna a la derecha de la pantalla sea cubierta por el marco, dejando sólo 39 columnas visibles en la pantalla. Como en el caso del registro 17, esta característica permite la inclusión de nuevos caracteres durante un scroll suave de la pantalla.

— BIT 4: Activa el modo multicolor cuando está a 1; éste permite la existencia de hasta 4 colores por carácter.

En este modo, sólo se permiten los 8 primeros colores en lugar de los 16 disponibles normalmente. Se examina para cada carácter a representar el valor almacenado en la RAM de color correspondiente. Si el bit 3 está a 0, el carácter se representa en multicolor. En este modo, cada dos bits de definición del carácter controlan un pixel de pantalla, o sea, que cada pixel es el doble de ancho. Según el valor de estos dos bits, el color del pixel será uno u otro:

00 — Registro color fondo 0
01 — Registro color fondo 1
10 — Registro color fondo 2
11 — Color según la RAM de color (los 3 bits menos significativos).

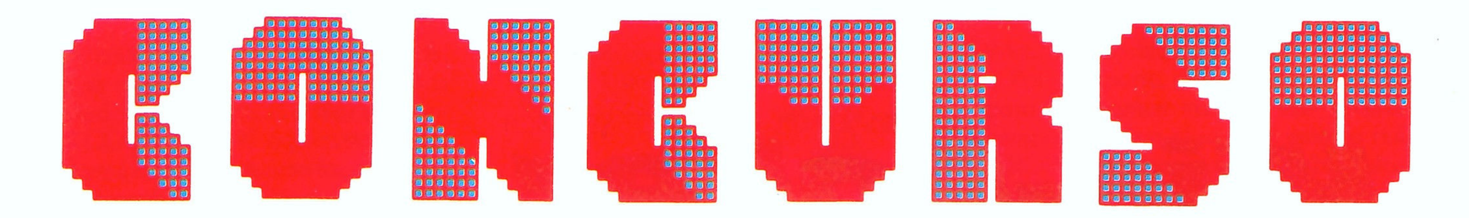
— BIT 5: Este bit controla la secuencia de reset del VIC. Si se pone a 1, se interrumpen todas las funciones del mismo, incluyendo el refresco de la RAM. Por tanto, no conviene ponerlo a 1, o de lo contrario...

— BIT 6, 7: Sin uso, suelen estar a uno, pero no cumplen ninguna función especial.

## Registro 23 (Dir. 53271)

Cada uno de los bits de este registro controlan la expansión vertical de los sprites. La expansión se activa poniendo a 1 el bit correspondiente (el 0 para el sprite 0, etc.).

Desgraciadamente, nuestro espacio no da más de sí. Continuaremos la descripción de los registros en el próximo número.






***El travieso C-Byte tiene el honor de invitaros a la participación en nuestro quinto concurso de programación. Los requisitos necesarios son bien pocos:***

- Saber programar un ordenador COMMODORE.
- Ser español o extranjero y
- Tener una edad comprendida entre 5 y 105 años.

Fácil, ¿verdad?
En cuanto a los premios, la mar de atractivos:

**dotado con un gran premio de**

**100.000 Ptas.**

**en material informático, a escoger por el premiado.**

Y en fin, si alguno de los programas destaca por su originalidad, estética o comicidad, no sería de extrañar que le cayera alguna cosilla más...

## BASES DEL CONCURSO

**1.** Los programas remitidos al concurso deberán ser creación original del autor o autores, y completamente inéditos, pudiendo remitir tantos programas como se desee.

**2.** Los programas deberán ser enviados en casete o disco flexible a TU MICRO COMMODORE (Concurso de programación). Apartado de Correos 61.294. 28080 MADRID.

**3.** Los programas podrán ser de cualquier tipo (juegos, utilidades, gestión, educativos) y habrán de estar escritos en lenguaje BASIC o Código Máquina.

**4.** Los programas deberán ser remitidos desprovistos de cualquier tipo de protección, que impida o dificulte el análisis del mismo, así como reproducción en estas páginas y su introducción como listado siguiendo el sistema FUERA ERRORES.

**5.** Cuando la ejecución del programa precise de la concurrencia de determinado periférico o aditamento (joysticks, tabletas gráficas, programas comerciales de ayuda), se valorará la indicación de las modificaciones pertinentes, para que el programa pueda ser disfrutado por cualquier usuario en la configuración básica.

**6.** Todo programa presentado al concurso deberá acompañarse de los siguientes datos:

- Datos personales del concursante.
- Nombre del programa.
- Modelo para el que está destinado.
- Breve descripción del programa detallando las indicaciones necesarias para su ejecución.

**7.** Los programas premiados pasarán a ser propiedad de la revista TU MICRO COMMODORE, pudiendo hacer ésta libre uso de ellos, y renunciando sus autores a cualquier otra compensación distinta al premio.

**8.** Los programas no premiados que por su calidad se hagan merecedores de su publicación, serán adquiridos por la editorial, aplicando la tarifa vigente.

**9.** Los programas recibidos con posterioridad a la fecha tope de la presente edición, serán automáticamente incluidos en los destinados a la siguiente.

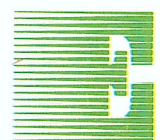
**10.** El jurado decidirá sobre todos los aspectos no contemplados en estas bases y su decisión será inapelable.

Y ahora a darse mucha prisa, el plazo para la recepción de programas termina el próximo día 1 de Marzo de 1987.

**¡¡SUERTE!!**

# Hijo-Poker

De apuestas va el programa... ¿Qué tal anda tu bolsillo? ¿Te precias de tener una buena visión de la jugada? Pues aquí está tu oportunidad: ¡apuesta al «hijo-poker» y pronto descubrirás cuán apropiado resulta el nombre!

l programa desarrolla un conocido juego español, aunque, en este caso, con baraja francesa. Se trata, ni más ni menos, de superar la carta que extraiga al azar el ordenador, con las cuatro cartas que te hayan tocado en suerte; desde luego, si con cualquiera de tus cuatro cartas eres capaz de superar a la del ordenador, el importe de la apuesta pasará a engrosar tu bolsa. La escala de valores o puntuaciones que determinan que carta predomina sobre otra es la siguiente:
A>K>Q>J>10>9>8>7>6>5>4>3>2

Y, por supuesto, para ganar a la carta extraida es necesario que el jugador posea una carta del mismo palo y puntuación superior.

Como suele ser habitual al tener al ordenador como compañero de juego, éste se ocupará de casi todo: barajará las cartas y hará el reparto, mostrando en la pantalla las cuatro cartas que han correspondido al jugador. Acto seguido, éste puede ya realizar su apuesta; y ¡ojo! que no se admiten trampas: las apuestas negativas serán consideradas positivas por el ordenador. Una vez cerrada la apuesta, el propio ordenador extraerá la carta oculta y comunicará a los presentes quién ha resultado vencedor. La pantalla visualizará el texto oportuno y se emitirán dos sonidos distintos según te haya sonreido o no la fortuna.

Antes de cada apuesta, el jugador puede observar sus disponibilidades monetarias en la parte superior de la pantalla. Por lo demás, el ordenador rechazará aquellas apuestas que excedan de la máxima permitida; ésta se define antes de empezar el juego, al tiempo que se establece el importe de cada jugada.

El funcionamiento del programa es simple. En primer lugar se construye la baraja y, a continuación, se redefinen los caracteres. Concluida esta primera etapa, se salta al bucle principal, que se repite indefinidamente.

Para introducir el programa a través del teclado es recomendable aprovechar las posibilidades que brinda el editor de pantalla del COMMODORE-64; las siguientes instrucciones en modo directo facilitarán el trabajo en gran medida. Veamos:

a) Introduce las siguientes instrucciones:

POKE 650,128 (autorrepetición de teclas).

FOR J=12 TO 24:PRINT J"N$(";J–10;")= C$+CHR$(C)+":NEXT

Ahora lleva el cursor al primer PRINT y mantén pulsada la tecla RETURN.

Con esta simple receta, el ordenador escribirá automáticamente casi la mitad de las sentencias de las líneas 12 a la 24. Por supuesto, luego debes volver sobre las mismas para completarlas.

b) Una instrucción más:

FOR J=27 TO 38:PRINT J"DATA":NEXT

Actúa de igual forma que en el caso anterior y te encontrarás con que ya tienes escritas parte de las instrucciones de las líneas 27 a 38.

c) Para escribir de forma automática la parte común de las líneas 80 a la 94, debes aplicar el referido método después de introducir:

FOR J=80 TO 94:PRINT J"DATA":NEXT

Con este recetario, el propio Commodore 64 ocupará el lugar de tus dedos, introduciendo una buena parte del programa... ¡Más facilidades, imposible!

Por otra parte, estos pequeños truquillos para ahorrarnos trabajo, son totalmente compatibles con el uso del sistema FUERA ERRORES, ya que una vez completadas las líneas que generemos de esta forma, al pulsar RETURN, debemos obtener el número clave adecuado, de la misma forma que si hubiésemos tecleado toda la línea.

Como ya sabemos que el juego no está bien, vamos a terminar dándole un toque didáctico, para lo cual aportaremos las tablas de la estructura del programa y uso de las variables. Seguro que nos ayudarán al entendimiento del programa, y podremos aprender de él algo más sobre programación.

## LISTADO

```
0 REM ** HIJOPOKER ** A. CARVAJAL                   -218-
1 S$="{4 ARB}{4 IZQ}":S=54272                       -207-
2 C$="{BLK}<A>!3 C!<S>{ABJ}{5 IZQ}!-![3 ESP]!-!{ABJ}{
5 IZQ}!-![3 ESP]!-!{ABJ}{5 IZQ}!-![3 ESP]!-!{ABJ}{5 I
ZQ}!-![3 ESP]!-!{ABJ}{5 IZQ}<Z>!3 *!<X>":C$=C$+S$
                                                    -020-
3 DIM CA(14,4),N$(14,4)                             -155-
4 S=54272:FORL=STOS+24:POKEL,0:NEXT:POKES+5,13:POKES+
24,15                                               -129-
5 FORJ=1TO4                                         -203-
6 IFJ=1THENP$="!A!":C=144                           -213-
7 IFJ=2THENP$="!X!":C=144                           -238-
8 IFJ=3THENP$="!S!":C=28                            -188-
9 IFJ=4THENP$="!Z!":C=28                            -197-
12 N$(2,J)=C$+CHR$(C)+"2{ABJ}"+P$+"{2 ABJ}{IZQ}"+P$
                                                    -112-
13 N$(3,J)=C$+CHR$(C)+"3{ABJ}"+P$+"{ABJ}{IZQ}"+P$+"{A
BJ}{IZQ}"+P$                                        -030-
14 N$(4,J)=C$+CHR$(C)+"4{ABJ}{IZQ}"+P$+" "+P$+"{2 ABJ
}{3 IZQ}"+P$+" "+P$                                 -169-
15 N$(5,J)=C$+CHR$(C)+"5{ABJ}{IZQ}"+P$+" "+P$+"{ABJ}{
2 IZQ}"+P$+"{ABJ}{2 IZQ}"+P$+" "+P$                 -087-
16 N$(6,J)=C$+CHR$(C)+"6{ABJ}{IZQ}"+P$+" "+P$+"{ABJ}{
3 IZQ}"+P$+" "+P$+"{ABJ}{3 IZQ}"+P$+" "+P$          -194-
17 N$(7,J)=C$+CHR$(C)+"7{ABJ}{IZQ}"+P$+" "+P$+"{ABJ}{
3 IZQ}"+P$+P$+P$+"{ABJ}{3 IZQ}"+P$+" "+P$           -213-
18 N$(8,J)=C$+CHR$(C)+"8{ABJ}{IZQ}"+P$+" "+P$+"{ABJ}{
3 IZQ}"+P$+P$+P$+"{ABJ}{3 IZQ}"+P$+P$+P$            -232-
19 N$(9,J)=C$+CHR$(C)+"9{ABJ}{IZQ}"+P$+P$+P$+"{ABJ}{3
 IZQ}"+P$+P$+P$+"{ABJ}{3 IZQ}"+P$+P$+P$             -251-
20 N$(10,J)=C$+CHR$(C)+"10"+P$+"{ABJ}{3 IZQ}"+P$+P$+P
$+"{ABJ}{3 IZQ}"+P$+P$+P$+"{ABJ}{3 IZQ}"+P$+P$+P$
                                                    -139-
21 N$(11,J)=C$+CHR$(C)+"J{ABJ}{IZQ}"+P$+"[]"+"{ABJ}{3
 IZQ}"+"!-!#!-!"+"{ABJ}{3 IZQ} $"+P$                -203-
22 N$(12,J)=C$+CHR$(C)+"Q{ABJ}{IZQ}"+P$+"&("+"{ABJ}{3
 IZQ}"+"!-!^!-!"+"{ABJ}{3 IZQ}+*"+P$                -119-
23 N$(13,J)=C$+CHR$(C)+"K{ABJ}{IZQ}"+P$+"@\"+"{ABJ}{3
 IZQ}"+"[3 ^]"+"{ABJ}{3 IZQ}')"+P$                  -190-
24 N$(14,J)=C$+CHR$(C)+"A{2 ABJ}"+P$+"{ABJ}"
                                                    -121-
25 NEXTJ                                            -016-
26 POKE53281,7:POKE53280,5                          -233-
27 PRINT"{CLR}{BLK}{ABJ}"TAB(15)"{RON}HIJOPOKER{ROF}"
                                                    -175-
28 PRINT"{ABJ}{DCH}EL OBJETIVO DEL JUEGO ES[2 ESP]GAN
AR[2 ESP]A[2 ESP]LA"                                -218-
29 PRINT"{DCH}CARTA QUE SACA EL ORDENADOR CON UNA DE"
                                                    -067-
30 PRINT"{DCH}LAS CUATRO NUESTRAS TENIENDO EN CUENTA"
                                                    -221-
```

```
31 PRINT"(DCH)QUE: A>K>Q>J>10>9>8>7>6>5>4>3>2."
                                                  -001-
32 PRINT"(ABJ)(DCH)Y QUE LA CARTA SEA DEL MISMO PALO.
"                                                 -045-
33 PRINT"(ABJ)(DCH)EN ESTE JUEGO NO HAY COMODINES."
                                                  -177-
34 PRINT"(ABJ)(DCH)TU DETERMINAS EL VALOR DE CADA[2 E
SP]JUGADA"                                        -078-
35 PRINT"(DCH)ASI COMO EL VALOR DE LA APUESTA MAXIMA"
                                                  -092-
36 PRINT"(DCH)Y NO SE ADMITEN APUESTAS NEGATIVAS."
                                                  -221-
37 PRINT"(ABJ)[11 ESP]ESPERA, POR FAVOR"          -103-
38 PRINT"[6 ESP]SE ESTAN CREANDO LAS CARTAS"
                                                  -073-
46 POKE52,48:POKE56,48                            -036-
47 POKE56334,PEEK(56334)AND254                    -212-
48 POKE1,PEEK(1)AND251                            -042-
49 FORI=0TO2023:POKEI+12288,PEEK(I+53248):NEXT
                                                  -011-
50 POKE1,PEEK(1)OR4                               -141-
60 POKE56334,PEEK(56334)OR1                       -051-
65 POKE53272,(PEEK(53272)AND240)+12               -170-
70 FORI=12288TO12295:READA:POKEI,A:NEXT           -206-
71 FORI=12504TO12543:READA:POKEI,A:NEXT           -194-
72 FORI=12568TO12639:READA:POKEI,A:NEXT           -211-
80 DATA 170,124,84,127,108,68,127,255             -104-
81 DATA 255,126,84,124,108,68,120,255             -101-
82 DATA 0,0,32,252,44,30,63,63                    -243-
83 DATA 64,96,112,120,88,24,24,24                 -155-
84 DATA 255,239,239,239,239,239,239,255           -230-
85 DATA 24,24,24,26,30,14,6,2                     -197-
86 DATA 60,60,60,60,60,60,60,60                   -044-
87 DATA 255,62,34,54,62,42,126,255                -212-
88 DATA 255,247,247,247,247,247,247,255           -228-
89 DATA 120,124,86,127,111,71,127,255             -098-
90 DATA 252,252,120,52,63,4,0,0                   -032-
91 DATA 0,24,60,60,60,24,24,24                    -242-
92 DATA 255,254,33,54,254,42,62,85                -213-
93 DATA 255,254,226,246,254,106,62,30             -100-
95 DATA 24,24,24,60,60,60,24,0                    -246-
450 INPUT"(CLR)(BLK)(ABJ)(3 DCH)PRECIO DE LA JUGADA";
PJ                                                -209-
460 INPUT"(ABJ)(3 DCH)APUESTA MAXIMA";AM          -175-
480 FOR TT=1 TO 2000:NEXT TT:T=9                  -195-
482 IF G>=0 THEN PRINT "(CLR)(BLU)(11 DCH)GANAS:"G"PE
SETAS"                                            -244-
484 IF G<0 THEN PRINT "(CLR)(RED)(10 DCH)PIERDES:"ABS
(G)"PESETAS"                                      -096-
490 FOR K=1 TO 5                                  -149-
500 X=INT(RND(8)*13)+2:Y=INT(RND(8)*4)+1:IF CA(X,Y)=1
 THEN 500                                         -030-
510 CA(X,Y)=1:P(K)=Y:V(K)=X                       -083-
515 NEXT K                                        -101-
517 FOR K=1 TO 4                                  -148-
520 PRINT"(HOM)(7 ABJ)"TAB(T);N$(V(K),P(K));S$;:T=T+6
                                                  -003-
525 NEXT K                                        -102-
526 INPUT "(HOM)(13 ABJ)(2 DCH)(BLK)APUESTA";A:A=ABS(
A):IFA>AM THEN 526                                -252-
530 FOR K=1 TO 4                                  -143-
540 IF P(5)=P(K) THEN IF V(5)<V(K) THEN JU=1:GOTO565
                                                  -030-
545 NEXTK                                         -072-
546 JU=-1                                         -249-
565 PRINT"(HOM)(14 ABJ)"TAB(18)N$(V(5),P(5))
                                                  -196-
570 IF JU=1 THEN PRINTTAB(16)"(ABJ)GANASTE!!":G=G+A-P
J:II=6E3:FI=16E3:GOSUB700                         -194-
580 IF JU=-1THEN PRINTTAB(15)"(ABJ)PERDISTE!!":G=G-A-
PJ:II=12E3:FI=2E3:GOSUB700                        -038-
600 FOR I=1 TO 14:FOR J=1 TO 4:CA(I,J)=0:NEXT J,I:GOT
O 480                                             -142-
700 FORM=IITOFISTEPJU*200:HF=INT(M/256):LF=M-HF*256:P
OKES+1,HF:POKES,LF                                -001-
710 POKES+4,17:NEXT:POKES+4,16:RETURN             -110-

READY.                                            -173-
```



## CHISPA:

En el diseño de pantallas de baja resolución, a menudo nos encontramos con el problema de escribir algo en la última posición de la pantalla, sin que toda ella haga el fatídico «scroll» que arruinaría el trabajo. Normalmente, esto se obvia mediante el uso de POKE a la posición deseada (2023 en el C64) con el correspondiente POKE a la memoria de color. Aunque esta técnica no es mala, vamos a presentar otras alternativas más elegantes.

```
10 PRINTCHR$(147)CHR$
   (5);:FORI=1TO24:FORJ
   =1TO40:PRINT".";:NEXT:
   NEXT
20 FORI=1To38:PRINT"*";:
   NEXT:PRINTCHR$(144)
   "U";
30 PRINTCHR$(5)CHR$
   (157)CHR$(148)"*";
40 GOTO40
```

Donde U es el último carácter de la pantalla en color negro (CHR$(147)). La técnica es escribir el último carácter en la posición 39 de la última línea, a continuación se efectúa un cursor izquierda (CHR$(157)) con el cambio al color original (CHR$(5)) nuevamente, con lo que el carácter 39 (nuestra U negra) pasa a ocupar la posición 40, escribiendo en la posición 39 el asterisco.

Para el C128 y el C16 hay otra posibilidad adicional: la de desactivar el scroll mediante PRINTCHR$(27);"M" (se vuelve a activar con PRINTCHR$(27);"L") o la de usar los POKES correspondientes que ejecutan la misma función: para C16, POKE2025,255 es Scroll Off y POKE2025,0 es Scroll On; las correspondientes en el C128 son: POKE 248,255 Scroll Off y POKE 248,0 Scroll On.

## CHISPA:

De todos los comandos y sentencias disponibles en el BASIC de nuestro C64, hay algunos que se utilizan muy a menudo y otros, en cambio, apenas son una reseña en el manual del usuario.

Entres estos últimos, vamos a referirnos a la función POS(). La verdad es que su uso no es muy espectacular, pero veamos para qué puede servir.

Como todos sabemos, POS() devuelve la posición del cursor en la línea lógica (0-79) en el momento de su utilización, entre los paréntesis irá un argumento ficticio (un número o una variable cualquiera). En el siguiente ejemplo utilizamos la función POS() para no cortar las palabras de un texto que va incluido en la variable A$ y se escribe, repetidamente, con GOSUB100.

En esta rutina, se escribe el contenido de A$, teniendo cuidado de que el cursor no haya sobrepasado un cierto límite C y esperando a encontrar un espacio para saltar de línea. Introduce como tabulación el número de caracteres libre por la izquierda y en corte texto el valor mínimo de margen por la derecha. Prueba con varios colores y comprenderás mejor el programa según los resultados:

```
  1 INPUT "TABULACION
    .:";T
  2 INPUT "CORTE TEXTO:
    ";C
  5 PRINTCHR$(147);TAB
    (40+T);
 10 A$="EL COMMODORE
    64 ES UN ORDENADOR
    MUY COMPLETO, DE
    GRAN RELACION": GO
    SUB 100
 15 A$="CALIDAD PRE
    CIO, ENTRE LAS CA
    RACTERISTICAS MAS
    NOTABLES": GOSUB
    100
 20 A$=" DESTACAN SU
    RESOLUCION DE
    320X200 PIXELS Y LA
    CAPACIDAD DE": GO
    SUB 100
 30 A$=" GRAFICOS MUL-
    TICOLOR EN CUATRO
    COLORES. EL SONIDO
    ES DIGNO": GOSUB
    100
 40 A$=" DE MENCION POR
    SU EXTRAORDINARIA
    FLEXIBILIDAD DE CON
    TROL,": GOSUB 100
 50 A$=" GRACIAS A LA
    MANIPULACION DEL
    ADSR. LOS SPRITES O
    FIGURILLAS": GOSUB
    100
 60 A$=" MOVILES, SON
    UN COMPLEMENTO
    PERFECTO.": GOSUB
    100
 80 END
100 FOR I=1 TO LEN(A$)
110 IFMID$(A$,I,1)=" "
    AND POS(0)>C THEN
    PRINT:PRINTTAB(T);: I=
    I+1
120 PRINTMID$(A$,I,1);:
    NEXT:RETURN
```

## CHISPA:

Alguna vez hemos hablado de lo que es el «garbage collection» o recogida de basura, consistente en liberar la memoria de cadenas de las que ya no se usan, habiendo mencionado que dicha operación puede tardar bastante tiempo. Aquí tienes un ejemplo de cuánto puede ser ese tiempo: cambia el valor de la variable X al principio y verás cómo cambian los tiempos.

```
10 X=500:DIMA$(X):FORI=
   0TOX:A$(I)=STR$(I):
   NEXT
20 PRINT"INICIA LA RECO
   GIDA":T=TI:Z=FRE(.):
   PRINT(TI-T)/60"SEGUN
   DOS"
```

## CHISPA:

En una chispa anterior decíamos que SYS59903 borraba de la pantalla una línea que previamente habíamos colocado en la posición 781 mediante POKE 781, línea. Usando esta técnica y otra de impresión, vamos a borrar la pantalla de dos formas espectaculares.

```
10 FORI=0TO23:B$=B$+
   " "+CHR$(157)+
   CHR$(17):NEXT:B$=
   B$+CHR$(145)
20 FORI=0TO19:IFI=0THEN
   POKE781,24:SYS59903
30 PRINTCHR$(19)TAB(I)
   B$:PRINTCHR$(19)TAB
   (39-I)B$:NEXT:PRINT
   CHR&(147),
```

Así borramos la pantalla desde los laterales hacia el centro por columnas; ahora veremos cómo borramos desde arriba y abajo hacia el centro por líneas.

```
10 FORJ=0TO12:POKE781,
   I:SYS59903:POKE
   781,24-I:SYS59903
20 FORK=0TO20:NEXT:
   NEXT:PRINTCHR$(147);
```

## CHISPA:

Si con el Commodore 128 quieres cambiar a modo Commodore 64, ya hay un SYS que lo hace, es: SYS65357; con esto no hace falta que teclees GO 64.

## CHISPA:

Para que no salga el carácter «?» en un INPUT, trabajando en el Commodore 128, usaremos los siguientes POKES:

```
POKE21,64
INPUT...
POKE21,0
```

## CHISPA:

Para cambiar el color de la pantalla al color del borde, al igual que ocurre cuando se carga o se graba un programa en el Datassette, se pueden usar los siguientes POKES directos:

POKE53265,11: Pantalla=borde

POKE53265,27: Situación normal

## CHISPA:

¿Sabías que si pulsas la tecla SHIFT del lado izquierdo del teclado a la vez que las dos teclas del cursor aparece el símbolo de pica? Los intrincados caminos del firmware...

# Interface RS-232

Para extraer auténtico partido de un ordenador, es necesario relacionarlo con el exterior, ya sea con periféricos como impresoras o plotters, o con líneas de transmisión de datos a través de modems. Un excelente sistema de llevar a cabo esta comunicación es a traves del interface RS-232.

A continuación enfocaremos dos aspectos diferentes del interface RS-232: en primer lugar, una información general del mismo y posteriormente, como manejarlo en el C-64/128.

## ¿Qué es un interface?

Un interface es un dispositivo que permite adaptar señales bajo unas normas estándar, denominadas protocolos, de forma que dos equipos que las sigan sean capaces de comunicarse entre ellos.

Hay una gran cantidad de estos protocolos, dentro de los cuales existen a su vez versiones y modificaciones. Lo cierto es que cada uno tiene sus ventajas e inconvenientes, según la aplicación a la que los destinemos.

El concepto de interface no es tan sencillo como puede parecer y hay varios niveles de aplicación de las normas estándar. Primero está la forma física de la interface, el conector y la repartición del cableado dentro del mismo; después, los niveles lógicos



de las señales de transmisión y de control y finalmente, toda la lógica de control que permite la transmisión/recepción sin errores.

Los interfaces se dividen de una forma genérica en dos tipos: serie y paralelo; las diferencias entre ellos son notables y cada uno tiene su aplicación específica.

Los interfaces serie, sirven para mandar los datos bit a bit de una forma ordenada y secuencial, economizando las transmisiones a larga distancia, ya que requieren poco cableado y se abaratan los costes de los amplificadores de mantenimiento (cada cierta distancia hay que amplificar la señal para evitar las pérdidas, siendo necesario un amplificador por cada línea de datos y control).

Por otro lado, los interfaces paralelo envían varios bits de datos de forma simultánea; son menos económicos pero son más fiables y permiten una mayor rapidez en la transmisión. Ideales para interconexión de periféricos y cortas distancias.

El problema de elegir un interface u otro, depende de los costes y el tipo de aplicación, ya que emplear un interface paralelo como el Centronics en una transmisión vía modem es inviable, por la propia naturaleza de la red telefónica.

## Introducción al interface RS-232

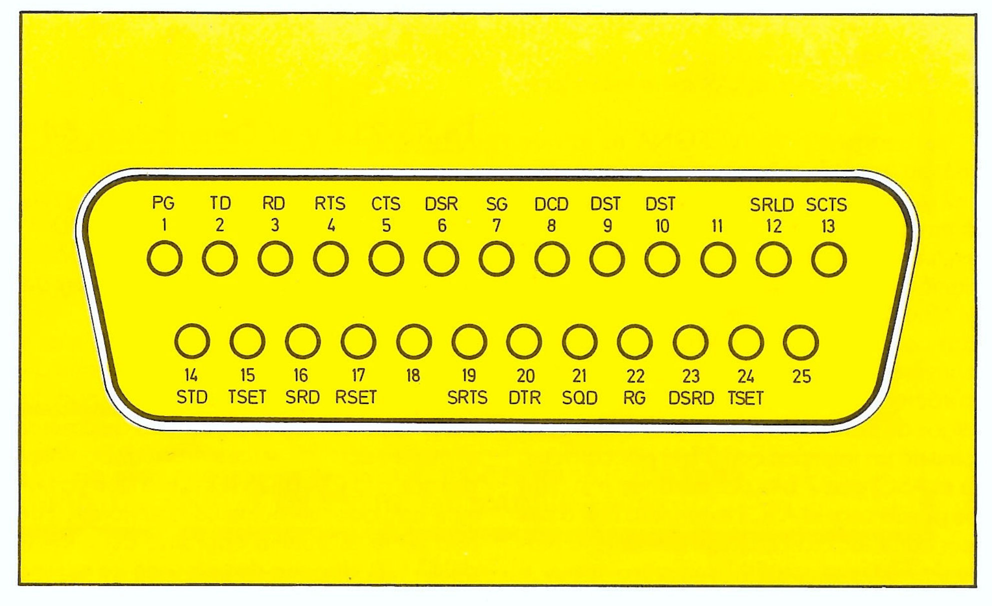
El interface RS-232 es una interface serie que permite conectar un equipo terminal (ordenador) con un periférico (impresora, plotter) o con un equipo de transmisión de datos a una red de telecomunicaciones (modem).

RS-232 es uno de los standard más extendidos en la actualidad, dada su versatilidad y facilidad de programación; por ello, tampoco define una comunicación por sí mismo ya que es necesario conocer una serie de parámetros previamente, como la velocidad de transmisión, la longitud de la palabra, etc.

Como es lógico, el control y la sincronización de una interface serie es mucho más complejo que el de un interface paralelo, pero esto queda resuelto con circuitos especializados que simplifican el trabajo, como es el caso de las VIAS (Versatile Interface Adapter) o las PIAS (Peripherical Interface Adapter), aunque el circuito más utilizado en el caso del RS-232 es la UART (Universal Asyncronus Receiver Transmiter).

Desde que la EIA (Electronic Industries Asociation) estableció el RS-232, han aparecido varias versiones del mismo: A, B y C, siendo esta última la más extendida y diferenciándose de las otras dos en que las señales están invertidas en su polaridad.

Este interface es tan versátil, que es capaz de conectarse sin problema desde un teletipo de baja velocidad, hasta una impresora láser de alta velocidad a un mismo ordenador y puede trabajar en una extensa gama de velocidades, tamaño de las palabras en bits y bajo distintas formas de operación.






Básicamente, puede funcionar de dos formas distintas, ASINCRONA o SINCRONA.

La transmisión SINCRONA utiliza una señal de reloj que emite la estación transmisora a intervalos regulares dependientes de la velocidad de transmisión (en baudios). De esta forma, la estación receptora sabe cuando se le van a enviar los datos, ya que estos van en bloques de cierta longitud, distribuidos de forma regular entre las señales de reloj.

La transmisión ASINCRONA es la que más se emplea; el transmisor envia una señal al receptor para que sepa que se le van a enviar datos y en ese momento se pone en funcionamiento el reloj para la sincronización de la transmisión.

Una de las características más destacadas de la RS-232 es la posibilidad de utilizar distinto número de bit para definir un carácter. Así es posible emplear varios códigos de transmisión, como el BAUDOT empleado en teletipos con 5 bits por carácter, o el ASCI con 7 bits por carácter e incluso se puede usar el ASCII extendido con 8 bits por carácter. En el caso de utilizar el formato de 7 bits por carácter, se suele emplear el último como bit de paridad para detectar errores en la transmisión.

Las tensiones de funcionamiento van de ±3V a ±12V y se varían según versiones.

Como hemos dicho antes, no sólo es necesario definir las señales de transmisión de datos, sino que son precisas una serie de señales de control, manejadas por todo el soporte lógico del protocolo de interconexión.

## La RS-232 y el Commodore 64

Los que hemos leído el manual de referencia del programador del COMMODORE 64, sabemos que éste dispone de una interface RS-232 C, accesible por el port de usuario.

Pero existen una serie de problemas que nos impiden utilizar el interface de forma directa. El RS-232 es accesible por el port de usuario, ya que todas las señales de control y transmisión se encuentran disponibles gracias a la CIA 6526 #2 (de la que ya hemos hablado en números anteriores). Por otra parte, el sistema operativo del C-64 y del C-128, dispone de una serie de rutinas Kernal encargadas de la gestión del RS-232. No obstante, a la hora de tomar las señales del port, nos damos cuenta que se trata de señales de niveles TTL de 0 a 5 voltios, las cuales no son utilizables bajo el estándar RS-232.

Para solucionar este problema, Commodore puso a la venta un cartucho RS-232 que compatibilizaba las señales al estándar. Un cartucho de este tipo consta de una fuente de alimentación auxiliar que nos proporciona las tensiones de +12V y de −12V y una serie de inversores y buffers que gestionan los niveles y polaridades de las señales.

En la interface RS-232 un «0» (Space) se indica por una señal negativa de −3V a −15V según el estándar y un «1» (Mark) con una señal positiva de +3v a +15V, donde toda señal por debajo de ±3V se considera ambigua y no es reconocida ni como Space ni como Mark.

## Descripción del patillaje del RS-232

De forma física, la interface serie RS-232

se organiza en torno a un conector «D» de 25 pines, de los cuales hay algunos que no están conectados y otros que se reservan para futuras expansiones.

1 (PG) Protective Ground: a este terminal está conectada la tierra de los equipos.

2 (TD) Transmit Data: este terminal está conectado al de recepción de datos del otro equipo colateral y es por el que se envian los datos.

3 (RD) Receive Data: este terminal está conectado al de transmisión de datos del equipo colateral y por el mismo se reciben los datos.

4 (RTS) Request To Send: por éste se envía la señal al equipo colateral con el cual se establece la comunicación.

5 (CTS) Clear To Send: es el terminal que contesta a la señal de Request To Send.

6 (DSR) Data Set Ready: este terminal indica cuando un equipo está listo para operaciones de I/O.

7 (SG) Signal Ground: a él están conectadas todas las masas de las señales de datos y de control, y es independiente del Protective Ground.

8 (DCD) Data Carrier Detect: detecta la presencia de portadora cuando trabajamos con el modem y sirve para localizar fallos en la transmisión de datos.

9 (DST) Data Set Testing.

10 (DST) Data Set Testing.

11 Sin asignar.

12 (SRLD) Secondary Receive Line Detector: como (RLD), pero para velocidades de transmisión superiores a 9600 baudios.

13 (SCTS) Secondary Clear To Send: como (CTS), pero para velocidades de transmisión superiores a 9600 baudios.

14 (STD) Secondary Transmited Data: igual que (TD), pero para velocidades de transmisión superiores a 9600 baudios.

15 (TSET) Transmisión Signal Element Timing: reloj para transmisión SINCRONA.

16 (SRD) Secondary Recived Data: es lo mismo que (RD), pero para velocidades de transmisión superiores a 9600 baudios.

17 (RSET) Received Signal Element Timing: reloj para recepciones SINCRONA.

18 Señal sin asignar.

19 (SRTS) Secondary Request to Send: es lo mismo que (RTS) pero para velocidades de transmisión superiores a 9600 baudios.

20 (DTR) Data Terminal Ready: esta señal se utiliza para indicar que el data set del dispositivo está listo para recibir y/o (depende del Request to Send) transmitir datos.

21 (SQD) Signal Quality Detect: indica (cuando está en off) que la portadora en la línea está en tal estado que probablemente habrá que repetir la transmisión.

22 (RG) Ring Indicator: se utiliza en MODEM's AUTOANSWER, para indicar llamada por vía telefónica.

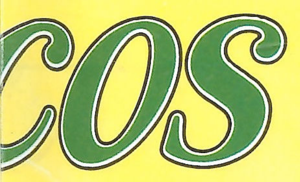
23 (DSRD) Data Signal Rate Detection: cuando en la línea está instalado un MODEM con dos velocidades, esta señal se utiliza para seleccionar una u otra.

24 (TSET) Transmit Signal Element Timig: por este fin se permite la entrada de un reloj externo para la velocidad de transmisión.

25 Señal sin asignar.

Confiamos en que los conocimientos aquí expuestos sirvan para un mejor entendimiento y aprovechamiento del interface RS-232.


LOS 7 MAGNIFIC
2. Uridium
6. Star S
7. Glass
5. Com
4. The Eidolon




Colaboremos todos en la confección de estos 7 MAGNIFICOS de TU MICRO COMMODORE. Envíanos el nombre de tu programa favorito, dejando muy claro tu nombre y dirección. Todos los meses, sortearemos cinco suscripciones por un año a nuestra revista entre las respuestas recibidas. Anímate y escríbenos a: TU MICRO COMMODORE. (7 MAGNIFICOS). Apartado de Correos 61.294. 28080 MADRID.

3. Little Computer People

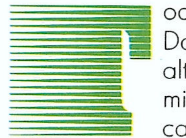
hip Andrómeda

mando

1. Green Beret

# Turb MICRO C-128

Si has adquirido un Commodore 128, pero no dispones de una unidad de discos, viéndote obligado a trabajar con tu Datassette, te interesará este artículo.



odos sabemos que el Datassette aporta una muy alta fiabilidad en el tratamiento de datos, aunque a cambio de una velocidad exasperantemente moderada, habiendo ocasiones, como la obtención de copias de trabajo de escasa utilidad, en las que este factor de fiabilidad no compensa la contrapartida de su lentitud.

Este problema generó la aparición de los programas turbo para casete, que aunque considerados como menos fiables que las rutinas incorporadas en la ROM, no son ni mucho menos inseguros. El problema es simple: a una mayor velocidad de transmisión existe una mayor posibilidad de error. Por otra parte, la solución a esta cuestión se hace también bastante obvia: encontrar la mayor velocidad posible siempre y cuando no sobrepasemos un punto crítico mínimo admisible, para considerar que el proceso se realiza con las suficientes garantias.

El principal inconveniente de los turbos es que dada su elevada velocidad, aparece una misteriosa incompatibilidad entre los Datassettes fabricados en Taiwan y los japoneses. Más exactamente, entre aquéllos aparatos que no tienen el cabezal de lectura/escritura ajustado al mismo nivel (azimut). Aunque ésta es una pega solucionable efectuando un reajuste mediante el tornillo a tal fin, situado encima del mismo cabezal, no es muy recomendable que dicha operación se lleve a cabo si se carece de experiencia en este menester.

No obstante, como pudimos comprobar en el artículo TUS MICRO ACCESORIOS (N.° 7, segunda época), existen en el mercado programas que nos ayudan en la corrección de las desviaciones del cabezal, incluso en forma de kits que incluyen cinta, destornillador de relojero apropiado para el diámetro del tornillo de azimut, etc...

En definitiva, los turbos ofrecen una buena solución a la parsimoniosa velocidad a que nos tiene acostumbrado el Datassette.

## Un turbo para el C-128

Es precisamente un turbo el programa que nos ocupa. Su denominación es: «TURB MICRO C-128» y funciona en el equipo C-128. En este mes presentamos el programa en sí, aunque su diseño permite extraerle aún más partido, quedando sus facultades más particulares para un próximo número.

Como facultad adicional, nuestro programa es compatible con algunos turbos del mercado existentes para C-64, como Fast Turb, Superturbo V3, Fast Tape, etc., lo cual permitirá intercambiar datos en turbo desde modo C-128 con Turb Micro C-128 a modo C-64, con cualquiera de sus compatibles y viceversa.

Una vez expuestas las características generales del programa, entraremos de lleno en él. Para comenzar, debe de introducirse el listado en BASIC que contiene el programa y grabarse. Seguidamente, se puede ejecutar RUN; en caso de que se hubiera cometido algun error, el programa nos informaría de ello. Si por el contrario, el listado ha sido introducido correctamente, obtendremos en la pantalla el mensaje:

TURB MICRO C-128.
READY.

Este es el indicador que nos informa que el programa está en memoria y funcionando. Siempre que pulsemos la secuencia RUN/STOP + RESTORE volverá a aparecer el mensaje, excepto si se corrige el vector $0A00-$0A01.

Antes de continuar, conviene recordar que la rutina FUERA ERRORES no es compatible con el modo C-128, de forma que no ha sido utilizada en el listado de nuestro programa. No obstante, éste incorpora una suma de comprobación, que detecta cualquier posible error en las líneas **DATA.**

El programa se encuentra en las posiciones de memoria $1300-$16A9, no siendo directamente reubicable; es decir, no es posible alterar la zona de carga de la rutina pretendiendo que ésta funcione correctamente.

## Opciones del Turb Micro C-128

Veamos ahora como extraer el máximo partido de todas las opciones que incorpora nuestro programa.

El sistema utilizado para diferenciar los diversos modos de operación, consiste en el uso del número de periférico (NP) y dirección secundaria (DS), a continuación del comando y nombre del fichero a tratar. Todos sabemos que el Datassette posee el NP 1, el canal RS 232 el 2, la pantalla el 3 etc. Nuestro turbo efectúa una reasignación de algunos de estos NP, pero sólo cuando se refieren a las instrucciones SAVE, LOAD y VERIFY. En cualquier otra, como OPEN o CLOSE, los NP no varían.

La tabla 1 prsenta las nuevas asignaciones.

El modo 1,F es el estandar de la ROM y su funcionamiento no resulta alterado por Turb Micro C-128. F es el denominado flag de carga (podemos encontrar más detalles en el manual del usuario.)

El modo 2,F es en el que se tratan los datos a velocidad fast y es necesario el programa Turb Micro C-128, o cualquier compatible en modo C-64, para poder leer los datos. Este modo funciona tanto desde BASIC como desde el monitor de código máquina, y por tener asignado el NP 2 resulta evidente su incompatibilidad con el canal RS 232.

El modo 3,0 se utiliza para grabar datos en modo fast, de tal manera que no resulta necesario disponer en memoria del turbo para poder leerlos. Sólo debe utilizarse desde modo BASIC.




Las direcciones 3,1 producen el mismo efecto que el 3,0 y además autoejecutan el programa tras su carga.

El modo 4,0 se utiliza para producir el mismo efecto que el 3,0, aunque en esta ocasión desde el monitor.

Es preciso advertir además, que las opciones 3 y 4 sólo son aptas para operar con el C-128, pero no se pueden utilizar desde el modo C-64.

A continuación, podemos ver un cuadro (tabla 2) con las diferentes operaciones a realizar con los comandos SAVE, LOAD y VERFI.

Naturalmente, tanto el nombre como el NP si es 1 o la DS si es 0 se pueden omitir, y en el monitor, las operaciones se deben realizar con sus correspondientes equivalentes S, L y V.

## Peculiaridades de Turb Micro C-128

Para finalizar, hablaremos de algunas pequeñas características del programa. La primera es el hecho ya comentado de que aparezca el mensaje "TURB MICRO C-128", cuando se pulse RUN/STOP y RESTORE. Esto nos informa que el programa sigue en funcionamiento, y en caso de que se produzca el mensaje «Ilegal Device Number» al intentar realizar alguna de las operaciones del turbo, es conveniente pulsar las teclas RUN/STOP + RESTORE para ver si el programa está funcionando; en caso negativo debe realizarse SYS 5612 para conectarlo.

Otra característica es que operando en el modo 2 se emite el código «BEL», para requerir nuestra atención cuando se presenta el mensaje «Found» o se finaliza la operación S, L o V en curso.

Asímismo, cabe mencionar el hecho que los nombres admitidos por la rutina tienen una longitud máxima de 15 caracteres, excepto en los modos 3 y 4, en que se extiende hasta 16.

Por otra parte, no será posible interrumpir los procesos de lectura o escritura mediante la tecla RUN/STOP, salvo en el punto de emisión del mensaje «Found», provocándose en este caso la salida al BASIC si nos encontrábamos en el monitor de código máquina.

No dudamos de la gran utilidad del programa presentado y remitimos a todos nuestros lectores interesados en el tema a próximas ampliaciones del mismo, que posibilitaron tratar directamente en turbo, pantallas de texto y gráficos, tanto en 40 como en 80 columnas, caracteres definidos por el usuario, así como definiciones de sprites. Todo ello, con la sencillez que proporcionan las ampliaciones de comandos.

**TABLA 1**

| NP | DS | FUNCION |
|---|---|---|
| 1, | F | FUNCIONAMIENTO NORMAL DEL DATASSETTE. |
| 2, | F | MODO TURBO. (SAVE-LOAD-VERIFY). |
| 3, | 0 | MODO TURBO CON CARGADOR BASIC. (SAVE). |
| 3, | 1 | MODO TURBO CON CARGADOR BASIC Y AUTORUN. (SAVE). |
| 4, | 0 | MODO TURBO CON CARGADOR MONITOR. (SAVE). |

**TABLA 2**

| NP | DS | SAVE | LOAD | VERIFY |
|---|---|---|---|---|
| 1, | F | Save"Nombre",1,F | Load"Nombre",1,F | Verify"Nombre",1,F |
| 2, | F | Save"Nombre",2,F | Load"Nombre",2,F | Verify"Nombre",2,F |
| 3, | 0 | Save"Nombre",3 | Load"Nombre",1 | Verify"Nombre",2 |
| 3, | 1 | Save"Nombre",3,1 | Load"Nombre",1, | Verify"Nombre",2 |
| 4, | 0 | Save"Nombre",4 | Load"Nombre",1 | Verify"Nombre",2 |

# LISTADO

```
10 PRINT"(CLR)(RVS)ESPERE UN MOMENTO."
20 FORX=0TO937:READA$:A=INPUT(A$):POKE4864+X,A:B=B+A:NEXT
30 IFB<>108154THENPRINT"(HOME)ERROR EN LINEAS DATA'S.":GOTO
PRINT"(CLR)";:SYS5612
100 DATA 03,26,03,28,03,20,20,20,20,20,20,20,20,20,20,20
110 DATA 20,20,20,20,20,56,E0,20,60,0B,20,98,0B,99,AC,00
120 DATA C8,C0,05,D0,F5,85,C6,20,60,0B,20,98,0B,20,BC,F7
130 DATA 45,90,85,90,20,C1,EE,20,B7,EE,90,EE,20,98,0B,45
140 DATA 90,85,90,58,08,20,9B,EA,20,57,EE,28,F0,03,4C,67
150 DATA 91,20,9B,F3,4C,BA,B3,8C,11,12,20,B5,5A,4C,90,AF
160 DATA 78,20,E4,EE,AD,11,D0,29,EF,8D,11,D0,20,74,E5,A9
170 DATA 07,8D,06,DD,20,A7,0B,26,BD,A5,BD,C9,02,D0,F5,A0
180 DATA 09,8C,30,D0,20,98,0B,C9,02,F0,F9,C4,BD,D0,E5,20
190 DATA 98,0B,88,D0,F6,84,90,60,20,5A,ED,20,A7,0B,26,BD
200 DATA C6,A3,D0,F7,A5,BD,60,A2,01,A9,10,2C,0D,DC,F0,FB
210 DATA AD,0D,DD,8E,07,DD,48,A9,19,8D,0F,DD,68,4A,4A,60
220 DATA 14,0B,20,B5,4B,A9,00,A0,0B,85,B2,84,B3,85,C0,AD
230 DATA 11,D0,A8,29,10,20,A1,16,98,29,6F,8D,11,D0,20,74
240 DATA E5,20,03,C4,E8,8E,30,D0,84,A4,84,90,78,60,A0,00
250 DATA A9,02,20,0C,14,A2,17,88,C0,09,D0,F4,A2,13,C6,AB
260 DATA D0,EE,98,20,0C,14,A2,16,88,D0,F7,60,85,BD,45,A4
270 DATA 85,A4,A9,08,85,A3,06,BD,A5,01,29,F7,20,2E,14,A2
280 DATA 24,EA,09,08,20,2E,14,A2,1F,C6,A3,D0,E9,60,CA,D0
290 DATA FD,90,05,A2,18,CA,D0,FD,85,01,60,20,89,14,C9,00
300 DATA F0,F9,85,AB,20,B3,14,91,B2,C8,C0,C0,D0,F6,F0,2A
310 DATA 20,89,14,20,B3,14,C4,93,D0,03,20,BC,F7,85,B0,20
320 DATA CC,F7,C5,B0,F0,04,A2,FF,86,90,45,A4,85,A4,20,C1
330 DATA EE,20,B7,EE,90,DD,20,B3,14,C8,84,C0,58,18,20,9B
340 DATA EA,20,57,EE,A9,07,4C,D2,FF,20,C8,E9,20,C2,13,A9
350 DATA 07,8D,06,DD,20,A7,13,26,BD,A5,BD,C9,02,D0,F5,A0
360 DATA 09,20,B3,14,C9,02,F0,F9,C4,BD,D0,E8,20,B3,14,88
370 DATA D0,F6,60,A9,08,85,A3,20,A7,13,26,BD,C6,A3,D0,F7
380 DATA A5,BD,60,85,93,A5,BA,C9,02,F0,03,4C,74,F2,A9,00
390 DATA 85,C6,C6,BA,20,3B,14,24,7F,30,03,20,F4,E8,20,B5
400 DATA 4B,A4,B7,F0,0C,88,20,AE,F7,D9,05,0B,D0,E6,98,D0
410 DATA F4,A6,AB,CA,F0,06,CA,F0,07,4C,D4,14,A5,B9,F0,13
420 DATA AD,00,0B,85,C3,AD,01,0B,85,C4,AD,04,0B,C9,10,B0
430 DATA 02,85,C6,20,33,F5,A5,C3,A4,C4,85,AC,84,AD,AD,02
440 DATA 0B,38,ED,00,0B,08,18,65,AC,85,AE,AD,03,0B,65,AD
450 DATA 28,ED,01,0B,85,AF,20,50,14,A5,BD,45,A4,05,90,F0
460 DATA 04,A9,FF,85,90,4C,9B,F3,A6,BA,E0,02,F0,13,90,0B
470 DATA E0,08,B0,07,E0,05,90,06,4C,4E,F5,4C,4E,F5,20,3D
480 DATA 16,A5,B9,29,01,85,B9,A9,02,85,AB,A5,C6,85,B0,20
490 DATA 51,ED,20,E9,E9,20,BC,F5,20,C2,13,20,EE,13,A5,B9
500 DATA 18,69,01,A2,13,20,0C,14,A2,19,B9,AC,00,20,0C,14
510 DATA A2,17,C8,C0,05,EA,D0,F2,A0,00,A2,05,20,AE,F7,C4
520 DATA B7,90,02,A9,20,A2,04,20,0C,14,A2,06,C8,C0,BB,D0
530 DATA EB,A9,02,85,AB,20,EE,13,98,A2,11,20,0C,14,84,A4
540 DATA A2,0D,8E,35,0A,20,CC,F7,AE,35,0A,20,0C,14,A2,06
550 DATA E6,AC,D0,04,E6,AD,A2,05,20,B7,EE,90,E5,A2,06,A5
560 DATA A4,20,0C,14,A2,15,88,D0,F6,4C,79,14,78,A9,32,A0
570 DATA 16,8D,00,0A,8C,01,0A,58,A9,C3,A0,14,8D,30,03,8C
580 DATA 31,03,A9,48,A0,15,8D,32,03,8C,33,03,A9,65,A0,FA
590 DATA 8D,14,03,8C,15,03,20,7D,FF,07,0D,2E,54,55,52,42
600 DATA 20,4D,49,43,52,4F,20,43,2D,31,32,38,0F,2E,00,4C
610 DATA 03,C4,20,E1,FF,D0,03,20,F8,15,4C,03,40,E0,03,D0
620 DATA 08,A5,B9,F0,04,A2,05,C6,B9,BD,95,16,8D,54,13,BD
630 DATA 98,16,8D,55,13,BD,9B,16,8D,56,13,A0,0E,20,AE,F7
640 DATA C4,B7,90,02,A9,20,99,05,13,88,10,F1,A5,C2,48,A5
650 DATA C1,48,A5,AF,48,A5,AE,48,A5,C6,48,A2,00,86,C6,A0
660 DATA 13,20,8A,E9,20,18,EA,A9,C0,85,C1,A9,C2,85,AE,20
670 DATA 18,EA,68,85,C6,4C,72,E9,4C,4C,8E,3F,BA,10,91,B3
680 DATA 12,2C,37,0A,30,03,8D,39,0A,60

READY.
```

# HOT LINE

**Nuestro amigo José M. Rey Clavero (VIGO), nos propone un par de preguntas, después de dar su crítica constructiva a nuestra revista. Por cierto, no te desesperes más con el programa PEPE LOTA, debido a un error ajeno a la redacción faltan líneas, aparecidas en el número 7.**
**¿Es posible llamar a comandos como RUN, LIST o NEW con una orden SYS? y ¿Existe alguna manera sencilla de programar esos comandos? o sea, algo parecido a la función KEY del SIMON'S BASIC.**

Pues sí, es posible hacerlo, en el caso de RUN sin embargo, necesitas dos llamadas consecutivas a dos rutinas de la ROM diferentes: la primera de ellas es SYS42585 y la siguiente SYS42926. Para LIST no conocemos ninguna, pero en código máquina es posible acceder a este comando colocando los caracteres LIST RETURN (5 en total) en el buffer de teclado y diciendo que hay 5 caracteres esperando así:

```
10 POKE631,ASC("L"):POKE632,ASC("I"):POKE633,ASC("S"):
   POKE634,ASC("T")
20 POKE635,13:POKE198,5
```

Ejecuta este pequeño programa y verás como se lista a sí mismo.
Para ejecutar NEW emplea SYS42564.
La manera más sencilla de programar las teclas de función, es por medio de interrupciones, aunque no es complicado en exceso, requiere un cierto número de pasos (no es sencillo, si con ello entendemos un SYS o similar), por lo extenso que sería una respuesta concreta, prometemos dedicarnos a ello en un próximo apartado de SOFTWARE.

---

**Nuestro amigo José González Gómez de Málaga, nos envía para tres preguntas que quedan extractadas como sigue:**
**«En chispas anteriores habéis publicado trucos para la desactivación de RUN/STOP, pero estos desaparecen al pulsar esa tecla junto con RESTORE».**
**«¿Cómo se puede hacer para que los INPUT tengan un aspecto más profesional, evitando los borrados de pantalla, desplazamientos de cursor y otros controles indeseables, sin tener que recurrir al uso de GET y chequeo ASCII?».**
**«¿Cómo es posible que algunos programas comerciales se autoejecuten al cargarse?»**

Para desactivar conjuntamente las teclas RUN/STOP RESTORE y además LIST como propina, puedes utilizar POKE808,225. Para volverlas a activar, teclea POKE808,237.
En cuanto a la evitación de fenómenos ciertamente molestos durante la entrada de caracteres en un INPUT, lo mejor es recogerlos (ya sabes, si no puedes con él...), para ello, lo mejor es utitilizar siempre INPUT con variables de cadena o alfanuméricas (terminadas en $) y transformarlas a numéricas si es preciso. Conseguir el efecto deseado, no es muy complicado si hacemos creer al ordenador que estamos en modo comillas, con lo que cualquier intento de atentar contra el programa, se convertirá en los consabidos caracteres de control en reverso. ¿Cómo se logra esto?, muy fácil, cambia los INPUT así:

```
POKE198,1:POKE631,34:INPUT "...";A$
```

Aparecerá una doble comilla antes de la entrada, pero ésta no será contabilizada como parte de A$.
Respecto a tu última pregunta, como el tema es interesante y extenso, lo trataremos pronto en el capítulo Software, pero como explicación para andar por casa, diremos que lo que hacen estos programas, es almacenar los datos necesarios para la ejecución directamente en la memoria, y se suele utilizar la zona 256-511, que es la página 1 de memoria, correspondiente al STACK del procesador, aquí se puede engañar al ordenador para que crea que ya se ha tecleado RUN.

**El problema que tiene nuestro amigo malagueño Juan Manuel Moyano Rodríguez, es la frecuente aparición del mensaje de error «out of data» al teclear los programas que aparecen en TECLA A TECLA.**

Como muy bien indica el texto del mensaje, este error se produce al intentar leer más datos de los disponibles dentro del

programa. Los datos están presentes dentro de las líneas DATA y son tratados por el intérprete como si fuesen una única lista SECUENCIAL.

Este mensaje puede aparecer por dos causas fundamentales: la primera es que el rango del FOR que los lee es superior al número de datos presentes en estas líneas DATA; y la segunda, es que no haya suficiente número de datos debido a que te has saltado alguna línea DATA. No obstante, también puede aparecer si has omitido alguna coma entre datos, lo cual produciría la confusión de dos o más datos como uno solo, faltando lógicamente al final.

Para solucionar este problema, deberás emplear el sistema. ¡Fuera Errores! siguiendo atentamente las instrucciones de uso que aparecen en cada número de nuestra revista, ya que parece altamente improbable que el ordenador genere estos mensajes por un mal funcionamiento.

Prueba a teclearlos con el ¡Fuera Errores! y si siguen apareciendo los mensajes, vuelve a escribirnos aclarándonos en qué línea se producen y en qué programas.

---

**Desde Barcelona nos escribe un jovencísimo amigo (9 años), David Ibáñez Alfonso, commodorero desde los siete, que nos plantea tres cuestiones:**

**En el programa «FUMAR» del número 1, tiene el problema nada agradable, de la desaparición súbita de la pantalla, a pesar de haber usado ¡FUERA ERRORES!**

**Al usar RESET y la chispa DESNEW, sobre el programa FUMAR, se recupera parcialmente el programa, fallando en las últimas líneas.**

**David ha observado que puenteando con la punta de una aguja, los pines 1 y 3 del port del usuario se ejecuta un RESET.**

Aprovechando tu carta, vamos a decir nuevamente que, debido a un error en el programa listador (CLISTO), algunos programas que empleaban cierto carácter, salían mal listados, por lo que hemos decidido volver a publicar los trozos erróneos (TU MICRO COMMODORE Noviembre). Aún así, puede que el error esté en la inversión de algunos números en las líneas POKE; esto daría lugar a la MISMA suma de control de ¡FUERA ERRORES! Repasa las líneas conflictivas y los listados rectificados anteriormente, entre las cuales se encuentra una línea absurda debida al fallo que hemos comentado.

Al utilizar este programa en concreto, interrupciones (alteraciones de los vectores normales), los resultados del DESNEW pueden ser imprevistos.

¡Cuidado!, esto es cierto, pero el peligro que se corre de tocar en algún lugar equivocado es grande, pudiéndose dañar el ordenador irremediablemente. Lo mejor es usar un montaje adecuado como el aparecido en TU MICRO COMMODORE número 2.

---

**Nuestro amigo Manuel Cortés, de Madrid, es un commodoriano reciente y tiene un par de dudas, al teclear nuestros listados. La primera de ellas es bien fácil: un símbolo que no comprende; se trata del acento cincunfrejo (repetimos: gorro de chino), que aparece en los listados como símbolo de «elevación a potencia» (a la derecha del asterisco).**

**La segunda cuestión es más conflictiva: no le caben las líneas de los listados ya que le ocupan más de dos líneas de pantalla.**

Por el ejemplo que nos pones en tu carta, deducimos que no has comprendido bien la filosofía del sistema ¡FUERA ERRORES! ni (todavía) la de los ordenadores Commodore.

Cuando dentro de comillas tecleas caracteres de control, como pueden ser colores, movimientos del cursor, funciones, etc., el C64 responde colocando unos caracteres en reverso, que indican la función que se realizará al imprimir; dichos caracteres son muy difíciles de interpretar desde las páginas impresas de una revista o incluso de un listado original. Por ello, y para facilitar las cosas, el sistema ¡FUERA ERRORES!, «descifra» estos caracteres y los cuenta por nosotros, siguiendo una nomenclatura preestablecida.

Tú, Manuel estás introduciendo las líneas tal cual aparecen en la revista y esto NO se debe hacer así. Por ejemplo, si encuentras (HOM) NO debes teclear esos cinco caracteres, en su lugar pulsa SOLAMENTE la tecla HOME, igualmente, al encontrar (3 ABJ) deberás pulsar 3 veces la tecla de cursor abajo (a la derecha del SHIFT derecho), etc. Asegúrate de que comprendes el sistema F.E., antes de teclear un programa (hay un artículo que los explica en cada número), y familiarízate con el teclado del C64 antes de continuar. Prueba ¡es fácil después de los mil primeros años!

---

**De nuevo, Juan Manuel Moyano Rodríguez (MALAGA), se dirige a nosotros: su problema ahora es que al grabar programas que le dejan los amigos, de casete a casete, con dos cables de intermediarios, los programas no cargan en su ordenador, aunque al ponerlos en un casete normal si se escuchan ruidos.**

El problema que mencionas, no se debe (probablemente) ni al datassette ni al ordenador, simplemente es un problema de fidelidad. Cuando el ordenador carga un programa, lo hace como una ristra interminable de unos y ceros que van a parar a la memoria. Estos unos y ceros están representados por medio de distintos sonidos audibles al reproducirlos.

En el caso de Commodore, y vistos al osciloscopio, estos sonidos se traducen en unas ondas complejas de mayor o menor amplitud, las cuales no son ni cuadradas ni sinusoidales; en base a ellas, el ordenador debe decidir qué es lo que está escuchando (1 ó 0). Al reproducir una cinta con dos cables, deberás asegurarte que lo haces en perfectas condiciones, es decir, con cables del tipo adecuado, con reproductores-grabadores de calidad, y ajustando correctamente el nivel de grabación (ni alto ni bajo). Aún así, la grabación no es segura ya que cualquier ruido no filtrado (que nosotros no notamos), puede convertir lo que originalmente era un 1 en un 0 (o dos ceros en un uno, o...), con lo cual oiremos que en la cinta hay «ruidos» grabados, pero... ¿entiende el ordenador esos ruidos?

---

**Escueta es la carta de Miguel Angel Ruiz (CORDOBA) que reproducimos íntegramente: «Cuando pulso las teclas RUN/STOP y RESTORE, con el programa «Fuera errores», éste no se desactiva. ¿Qué ocurre? ¿Es del ordenador? ¿Hay alguna otra forma de desactivar el programa? ¿Cómo?».**

Cuando se pulsan las teclas RUN/STOP RESTORE simultáneamente, el ordenador envía una señal de interrupción hardware directamente a una CIA (y de ahí a la CPU); esto equivale a decir al ordenador que detenga toda actividad y devuelva el control directo. Todo ello está muy bien, pero resulta que el ordenador no dedica todo el tiempo a chequear la tecla RESTORE, que de tecla no tiene nada (con perdón), ya que está conectada directamente a una de las CIAs. Por eso, para producir la interrupción, no basta con pulsar dicha tecla (repito, que no es tal), sino que hay que dar uno o varios golpes secos sobre ella, hasta conseguir el efecto deseado, es decir, que el ordenador se dé por enterado. Así pues, para desactivar F.E., insiste sobre esta tecla, a no ser que esté inhabilitada por software.

# ...Te seguimos presentando el mejor software del año




Con DANDY vivirás la aventura más complicada que jamás te hayas pensado en una mazmorra. No te será fácil encontrar el tesoro. DANDY es la mazmorra definitiva.

CSA




El universo, objetos tridimensionales, criaturas extrañas y la oscuridad del espacio, llenan de emoción y tensión este juego, donde tu supervivencia depende de tus reflejos.




CHAMPIONSHIP
BASKETBALL™
Two-on-Two™
GAMESTAR

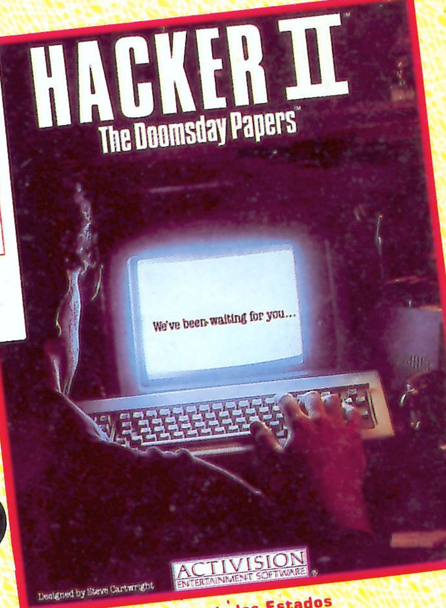



PRODIGY nos introduce, en el mundo "MEC" donde debemos conducir a "SOLO" el hombre sintético que cuida de "NEJO" y librarlo de los peligros más adversos, sin olvidarnos de WARDLOCK, el ser mecánico que quiere destruir toda vida orgánica. Sus efectos sonoros y en tres dimensiones lo hacen inmejorable y diferente.

CSA

Este BASKET empieza donde otros acaban, porque se basa en el juego de equipo. Con los mejores gráficos de un juego de BASKET y la variedad de posibilidades de juego (Dos jugadores y la computadora, prácticas, liga de 23 jugadores etc.) nunca te cansarás de jugar hasta llegar a ser un campeón.

CSA




Saludos del gobierno de los Estados Unidos... La CIA cuenta contigo para proteger a los países de Occidente. Los Rusos tienen en su poder el libro llamado "El día del juicio final". Con él pueden tener al mundo en sus pies. Y aquí entras tú, de lo demás, sólo podemos decirte: BUENA SUERTE. Falta te hace.

CSA

Disponibles para:
COMMODORE
SPECTRUM
AMSTRAD CASS/DISK
C S A

EN TIENDAS ESPECIALIZADAS Y GRANDES ALMACENES, O DIRECTAMENTE POR CORREO O TELEFONO A: **PROEIN, S.A.**
Distribuido en Cataluña por: DISCOVERY INFORMATIC C/. Arco Iris, 75 - BARCELONA - Tels. 256 49 08 / 09
Velázquez, 10 - 28001 Madrid - Tels. (91) 276 22 08/09